

**The Opus**  
**Computer-Based Conversation System**

**TECHNICAL REFERENCE MANUAL**

**by Trev Roydhouse**

**Private Printing**

**Version 1.70**

## The Fine Print

The information in this document is subject to change without notice. The information in this document does not represent a commitment on the part of anybody.

No author, programmer, or anybody else shall be liable for technical or editorial errors or omissions made herein; nor for incidental or consequential damages resulting from the furnishing, performance, use, or inability to use this material.

If you are a commercial software developer and want to use any of the ideas you find here, you should also have the decency to mention the source of "your" idea. This includes (but isn't limited to) a certain commercial BBS system that recently came up with the idea of having a smart interface between its BBS and end-user modem software ... which is amazingly similar to Opus's *FTB*.

Copyright 1991, Wynn Wagner III  
All rights reserved.  
Printed in the United States of America.

First edition (14 July 1991)

Second edition (20 February 1992, ever-so-slightly revised by Wynn Wagner III)

**Opus-CBCS**

Version 1.70

Copyright 1986, 1987, 1989, 1990, 1991, Wynn Wagner III  
All rights reserved

The software for Opus V 1.70 was written by Doug Boone  
with William Beebe, Ulf Nilsson, Trev Roydhouse and  
John Valentyn





**Opus V1.7**  
is dedicated to

**TOM JENNINGS**  
who thought up hobbyist e-mail

and

**WARD CHRISTENSEN**  
who thought up BBS'ing and XModem

TO THE  
HONORABLE  
MEMBERS OF THE  
HOUSE OF REPRESENTATIVES  
IN SENATE  
WASHINGTON, D. C.  
JANUARY 1, 1900

## Table Of Contents

1	INTRODUCTION.....	1
2	CONTROL FILE PARAMETERS.....	3
	2.1 DOS PATH SPECIFICATION .....	3
	2.2 DOS FILE SPECIFICATION .....	3
	2.3 OPUS TEXT FILE SPECIFICATION.....	3
	2.4 OPUS PRIVILEGE LEVEL NAME.....	3
	2.5 OPUS KEY/LOCK SPECIFICATION .....	3
	2.6 OPUS SECTION SPECIFICATION .....	4
	2.7 DOS ERRORLEVEL SPECIFICATION .....	4
	2.8 OPUS CONTROL FILE PARAMETERS.....	5
3	OPUS MENU SYSTEM.....	75
	3.1 MENU MAINTENANCE.....	75
	3.2 THE MENU FILE .....	77
4	OPUS EMBEDDED COMMANDS.....	89
	4.1 TASK NUMBER SUPPORT .....	89
	4.2 LANGUAGE SUPPORT .....	89
	4.3 OEC (BBS/GBS) FILES.....	90
	4.4 EMBEDDED COMMANDS DICTIONARY .....	91
	4.5 FULL TILT BOOGIE COMMANDS.....	96
5	OPUS SUPPORT FILES.....	99
	5.1 OPUS LOGIN SEQUENCE.....	103
6	COMMAND LINE SWITCHES.....	105
7	EXTERNAL PROGRAM USAGE.....	107
	7.1 INVOCATION FROM MENUS .....	107
	7.2 INVOCATION FROM EMBEDDED COMMAND FILES .....	109
	7.3 RELOADING LASTUS##.DAT .....	109
	7.4 EXTERNAL FILE TRANSFER PROGRAMS.....	110
8	EXITS AND ERRORLEVELS .....	113
	8.1 PREDEFINED ERRORLEVELS .....	113
	8.2 EVENTS.....	114
	8.3 EXTERNAL MAILER .....	114
	8.4 MENU SELECTIONS .....	114
	8.5 PRECEDENCE OF EXITS .....	115
9	ERROR MESSAGES.....	117
10	MATRIX SCRIPT KEYWORDS .....	119
11	LOG STRUCTURE.....	121
	11.1 THE LOG FILE .....	121
	11.2 LOG LINE ENTRIES .....	121
	11.3 STANDARD LINE ITEMS.....	122
	11.4 LEVELS OF LOGGING .....	122
	11.5 OOMP LOG MESSAGES .....	123
	11.6 OPUS LOG MESSAGES .....	123
12	USING DESQVIEW .....	129
	12.1 FOSSIL DRIVERS AND DESQVIEW .....	129
	12.2 DESQVIEW WINDOW SETUP .....	129
	12.3 TUNING .....	130
	12.4 FILE SHARING AND SEPARATION .....	130
13	USING DOUBLEDOS .....	133
	13.1 FOSSIL DRIVERS AND DOUBLEDOS.....	133
	13.2 MEMORY CONSIDERATIONS .....	133
	13.3 DOUBLEDOS SETUP.....	134
	13.4 OUTSIDE PROGRAMS .....	135
	13.5 PERFORMANCE .....	135
	13.6 FILE SHARING AND SEPARATION .....	136
14	OPUS UTILITIES.....	137
	14.1 CATALOG OF OPUS UTILITIES.....	137

1942

1. The first part of the report  
deals with the general situation  
of the country and the  
state of the economy.

2. The second part of the report  
deals with the results of the  
survey and the conclusions  
drawn from it.

3. The third part of the report  
deals with the recommendations  
made by the committee.

4. The fourth part of the report  
deals with the conclusions  
drawn from the survey.

5. The fifth part of the report  
deals with the conclusions  
drawn from the survey.

6. The sixth part of the report  
deals with the conclusions  
drawn from the survey.

7. The seventh part of the report  
deals with the conclusions  
drawn from the survey.

8. The eighth part of the report  
deals with the conclusions  
drawn from the survey.



# 1

## INTRODUCTION

... this book was not so much written as gutted out.

-- a very rich, famous author

The Opus CBCS Technical Reference Manual is intended to be a convenient place to look for technical information on Opus V1.70. The information contained in this manual is exhaustive, sometimes tedious, and perhaps incredibly boring. It was not written to entertain, but rather to be the absolute source of detailed answers that an experienced sys-op might need to take full advantage of the Opus CBCS V1.70 system.

General information needed for daily operations is contained in the Opus Operations Manual. In some instances, brief overview information is presented in the Operations Manual with a few examples. The complete lists and technical explanations on a given topic are usually to be found in this manual. As an example, embedded codes are explained in the Operations Manual with examples, but the complete listing of all the codes is contained in this manual. If you are not sure where to find something, please refer to the Table of Contents.

No document would be complete without giving credit to those who contributed to the effort. First and foremost, I want to thank BOB DAVIS. Bob wrote the original Opus Technical Reference Manual for Opus CBCS V1.13. It is no exaggeration to say that it was, and is in this substantially revised incarnation, the most significant documentation ever written for Opus. I also want to thank DOUG BOONE. He filled in the gaps in my knowledge of the ins and outs of Opus, put up with my frequent postings of what I perceived to be problems, and did not despatch Guido "downunder" to explain things to me in person.

Please send any corrections or suggestions to me at:

Fidonet 3:711/401 (PEP)  
FidoNet 3:711/501 (V32)

International FAX +61-2-887-1702  
Domestic FAX (02) 887-1702

PO Box 1605, Macquarie Centre, NSW 2113, Australia.

Trev Roydhouse, 11 July 1991.

# THE NEW YORK PUBLIC LIBRARY

ASTOR LENOX TILDEN FOUNDATION  
455 FIFTH AVENUE  
NEW YORK, N. Y. 10018

Open daily from 10:00 a.m. to 5:00 p.m.  
Closed on Sundays and public holidays.

For information and reservations, call (212) 854-5500.

The New York Public Library is a non-profit organization that provides access to information and knowledge for all. It is the largest library system in the United States, with over 50 million volumes and 30 million digital items. The library is a vital resource for the community, providing a place for learning, research, and cultural enrichment.

The library's collections include books, periodicals, microfilm, and digital resources. It also offers a variety of programs and services, including reading programs, research assistance, and cultural events. The library is committed to providing a welcoming and inclusive environment for all users.

The library's building is a landmark in New York City, and its collections are a treasure trove of knowledge and information. The library is a place where everyone can find what they need to learn, grow, and thrive.



**2****CONTROL FILE PARAMETERS**

This section is a detailed reference on all parameters that may appear in the Opus control file for use by NACL, the Opus control file compiler.

---

**2.1 DOS PATH SPECIFICATION**

Up to 39 characters defining a legal DOS path specification terminated with a '\.

---

**2.2 DOS FILE SPECIFICATION**

Up to 39 characters defining a legal DOS file specification, including extension. Any exceptions to this will be noted with the individual parameter definition.

---

**2.3 OPUS TEXT FILE SPECIFICATION**

Same as a DOS FILE SPECIFICATION without extension. This actually specifies two files. The first contains no ANSI graphics and is of the form FILENAME.BBS. This file may also contain AVATAR screen manipulation embedded codes (see section 4 of this manual) which will only be sent to users who have enabled the Avatar video option. The second file is of the form FILENAME.GBS and contains ANSI graphics.

---

**2.4 OPUS PRIVILEGE LEVEL NAME**

These privilege levels are specified using certain abbreviations. It is very important that you use these abbreviations exactly as listed. They are listed in order of precedence, highest to lowest.

```
Hidden
Sysop
Asstsysop
Clerk
Extra
Favored
Privil
Worthy
Normal
Limited
Disgrace
Twit
```

Hidden is special, in that no user can get to any entity that is marked as Hidden. Also, a user's access privilege may be set to Hidden; Opus will allow them to log on, but will drop carrier as soon as they enter their password.

---

**2.5 OPUS KEY/LOCK SPECIFICATION**

In addition to allowing the sysop to tailor access by the use of the twelve privilege levels (see above), Opus also allows the use of the "keys and locks" concept.

There are thirty-two different individual locks and keys defined by the uppercase (capital) letters A-Z and the decimal numbers 0-5. Any combination of locks and keys is possible which gives something like  $2.6313 * 10E35$  possible combinations.

If some access feature specifies a lock(s), then Opus will not let the user gain access unless the user has the corresponding keys turned on in the user's record in the Opus user file.

Default key(s) may be turned on automatically for all new users when they first logon, see the LOGON KEYS option below. The sysop may also turn on key(s) for a user who is online by hitting the K key and then pressing the relevant A-Z or 0-5 keys. Press ENTER to end the key setting sequence. Or a sysop can put embedded commands in a file to turn on and off key(s) automatically.

---

## 2.6 OPUS SECTION SPECIFICATION

In addition to allowing the sysop to tailor access by the use of the privilege levels (see above) and keys/locks (see above), Opus also allows the use of the "sections" or "sigs" concept. Sections may be setup in advance by the sysop or user defined.

There are thirty-two different sections available defined as the uppercase (capital) letters A-Z and the decimal numbers 0-5.

---

## 2.7 DOS ERRORLEVEL SPECIFICATION

When Opus exits, it will do so with a DOS errorlevel that can be checked in a DOS batch file (refer to your DOS reference manual). Certain errorlevels are reserved by Opus. In Opus control file and event file, you may specify exit errorlevels for various actions. All valid DOS errorlevels may be used except 0-4, and 255 (see further section 8 of this manual).

## 2.8 OPUS CONTROL FILE PARAMETERS

The Opus control file defines the layout and functionality for your unique BBS. The control file contains these sections:

System Section  
Language Section  
Equipment Section  
Matrix and Echomail Section  
Session Section  
Message and File Area Definitions  
Menu Section

These are described in great detail in sections 2.8.1 through 2.8.7. As the menu section is rather complex, it is treated separately in its own major section in this manual.

### 2.8.1 SYSTEM SECTION

Definition: Specifies the beginning of the System Section of the control file.

Required: Yes, at the beginning of the control file.

#### DOS CLOSE STANDARD FILES

Definition: .....Specifies that Opus will not use certain standard file handles (STDAUX and STDPRN) thereby freeing them for use by OOMP, the internal Opus echomail scanner.

Required: .....Maybe. Opus V1.70 will scan to 12 nodes at a time if this option is specified, 10 if it is not specified. It will speed the echomail processing up if you are scanning to more than 10 nodes, but only in the right circumstances. For example, if you were scanning an echomail conference to 11 or 12 nodes it would make a difference, but at 13 there would be no advantage. There would again be an advantage if you were scanning to 21, 22, 23 or 24 nodes.

#### END SYSTEM SECTION

Definition: .....Specifies end of the system section of the control file.

Required: ..... Yes

#### HIGH FILE <number>

Definition: .....Specifies the highest file area used by Opus. This is a used to terminate such commands as F Locate. Opus will also restrict the areas scanned to those which are in the user's "section" (if any).

Range: ..... 1-32767

Required: .....No, defaults to 50 if not specified.

Example: .....HIGH FILE 83

**HIGH MESSAGE <number>**

Definition: .....Specifies the highest message area used by Opus. This is used to terminate such commands as MSG\_SCAN. For example, if high message is set to 50, then MSG\_SCAN will stop scanning after area 50. Opus will also restrict the areas scanned to those which are in the user's "section" (if any).

Range: .....1-32767

Required: .....No, defaults to 50 if not specified.

Example: .....HIGH MESSAGE 75

**INDEX**

Definition: .....OBSOLETE. Opus V1.70 always uses the lookup index file.

**LOCKS**

Definition: .....Specifies whether Opus will use file locking on the user file (requires that DOS SHARE.EXE be loaded). Supported under DOS 3.1 and later only. Locking is used for multi-line Opus installations, to prevent corruption of the user file by simultaneous access between Opus systems.

Required: .....No, but may be needed for some multi-line Opus systems.

**LOG FILE <filename>**

Definition: .....Specifies filename used by Opus to log various events. This log gives a history of what has happened on the system, for example, user activity, scheduled events and mail transactions.

Range: .....DOS file specification.

Required: .....Yes

Example: .....LOG FILE c:\opus\opus.log

**LOG MODE <mode>**

Definition: .....Specifies how much detail is placed in the Opus log file.

Range: .....There are actually six levels of detail for the Opus log file. You determine the level of detail by placing one or more of the log mode options in the control file. It is recommended that you enable the most detailed log and run your system that way for a day or so, to see the level of detail available. Then change to the less detailed level that best suits your needs. The log mode ranges are:

Kind	Example	Description
NUL	LOG MODE NULL	no log
DEFAULT	(no log mode specified)	scanty log
TERSE	LOG MODE TERSE	minimal log
VERBOSE	LOG MODE VERBOSE	moderate log
TRACE	LOG MODE TRACE	copious log
	LOG MODE VERBOSE TRACE	intense log



**MONITOR HEIGHT <number>**

Definition: .....Defines the number of lines displayed on the sysop's local console. Works only for VIDEO DOS or VIDEO FOSSIL. For VIDEO IBM, 25 lines is fixed.

Range: .....8-99

Required: .....No

Example: .....MONITOR HEIGHT 43

**MONITOR WIDTH <number>**

Definition: .....Defines the number of columns displayed on the sysop's local console. Works only for VIDEO DOS or VIDEO FOSSIL. For VIDEO IBM, 80 columns is fixed.

Range: .....20-132

Required: .....No

Example: .....MONITOR WIDTH 100

**MULTILINE CHAT**

Definition: .....Specifies that Opus V1.70 has multiline chat capability. Opus will show the status of "Block chat" at the CONFIG menu.

Required: .....No

**MULTITASKER <kind>**

Definition: .....Specifies whether a multitasker is used and which one. Opus has code which reduces its overhead, while idle by giving up its time slice, if the appropriate multitasker is specified. Opus will also create the INMAIL##.\$\$\$ flags in the STATUS directory if a multitasker is specified (and is detected by Opus) or if the new Opus V1.70 LAN option is set.

Required: .....No, but recommended if one of the listed multitaskers is used. Opus automatically detects if DoubleDOS or DESQview is absent, so it is allowable to specify these options, even if you do not run Opus under DoubleDOS or DESQview all the time.

Kinds .....

Kind	Example
DeskView	MULTITASKER DESQVIEW
DoubleDOS	MULTITASKER DOUBLEDOS
LAN	MULTITASKER LAN
TaskView	MULTITASKER TASKVIEW

**NAME <name>**

Definition: .....Specifies the name of your BBS.

Range:.....Up to 45 seven bit characters.

Required: ..... Yes

Example: .....NAME The Fireside Opus Conversation Pit

**PATH MISC <path\>**

Definition: .....Defines path to Opus miscellaneous text files.

Range:.....DOS path specification.

Required: ..... Yes

Example: .....PATH MISC c:\opus\misc\

**PATH SPANN <path\>**

Definition: .....Defines path for SPANN#.BBS files (Special Announcement files)

Range:.....DOS path specification.

Required: ..... Yes, if special announcement feature is used.

Example: .....PATH SPANN c:\opus\spann\

**PATH STATUS\_DIR <path\>**

Definition: .....Specifies the path Opus V1.70 will use to place the following files (this was the INMAIL\_DIR in Opus 1.1x.):

- the temporary flag file, INMAIL##.\$\$\$ , generated by Opus when it is processing mail. INMAIL##.\$\$\$ is ONLY generated if you have a multitasker flag set in your PRM file. If you are not running a multi-line system, there is no reason for Opus to generate this file;
- the flag file ACTIVE##.DAT, created when a user is currently logged into the system;
- the LASTUS##.DAT file, created when a user logs on;
- the CHAT## file that contains interline messages.

Range:.....DOS path specification.

Required: ..... Yes, if you do not specify a STATUS\_DIR path, Opus will put the INMAIL##.\$\$\$ , ACTIVE##.DAT, CHAT## and LASTUS##.DAT files in your SYSTEM directory.

Example: .....PATH STATUS\_DIR c:\opus\flags\



### **PATH SYSTEM <path\>**

Definition: .....Defines the path to the Opus V1.70 area and files database definitions.

Range: .....DOS path specification.

Required: ..... Yes

Example: .....PATH SYSTEM c:\opus\

### **PATH TEMP <path\>**

Definition: .....Defines path for temporary files used by Opus for such things as message uploads. If you have a RAM disk, it is a good idea to point TEMP to it.

The TEMP path is fairly important if you allow such things as uploading messages. Because of the nature of uploads, you should make this a path that does not contain anything sensitive. In other words, this sub- directory is "fair game".

Range: .....DOS path specification.

Required: ..... Yes, if message uploading is allowed.

Example: .....PATH TEMP c:\opus\temp\

### **REBOOT**

Definition: .....A security feature. Specifies whether Opus should turn on the watchdog function in the fossil driver, when a user selects a function that will cause Opus to exit with an error-level. By turning on this function, the fossil will cause a system reboot, if the caller connection should be broken while Opus is exited. The reboot prevents another caller from gaining access to the function that was in progress when the carrier was lost. Otherwise you have to know that your external program will return to Opus properly and let Opus handle the loss of carrier. Can be disastrous on multi-line systems!

Required: .....No

### **SNOOP**

Definition: .....Enables snoop mode automatically when user calls the system, allowing sysop to view all BBS activity on his local screen.

Required: .....No Note that sysop can turn on snoop mode from his keyboard when caller is online. If, however, Opus exits, SNOOP will return to the state defined in the control file. If the sysop wishes SNOOP to be on all the time, it should be enabled in the control file.

### **SYSOP <name>**

Definition: .....Specifies the name of the sysop

Range: .....Up to 40 seven bit characters comprising one or two strings separated by spaces.

Required: ..... Yes

Example: .....SYSOP Guido

**TASK <number>**

Definition: .....Specifies task number to help prevent file conflicts in multi-line Opus installations. The task number is used to create unique filenames. Note that the task number is specified as decimal in the control file, but it will be written as hexadecimal when Opus creates task numbered files. For example, for a task number of 10, Opus would create LASTUS0A.DAT and INMAIL0A.\*\*\* files.

Range: .....1-16 (01H-0FH)

Required: ..... Yes, if a multitasker is specified, otherwise defaults to zero.

Example: ..... TASK 2

**USES COMMON <filename>**

Definition: .....Defines filename for storage of common data used by multiline Opus systems. Currently this includes total user calls to all systems, a common quote file pointer, the last User ID used and the last message identifier used. This file will be created by Opus when required, provided you have enabled this option in the control file.

Range: .....DOS file specification.

Required: .....No

Example: ..... USES COMMON c:\opus\common.dat

**USES PASSWORD <filename>**

Definition: .....Defines filename for user file.

Range: .....DOS file specification.

Required: ..... Yes (see example below for "conventional" filename)

Example: ..... USES PASSWORD c:\opus\user.dat

**USES SCHEDULE <filename>**

Definition: .....Defines filename for schedule file.

Range: .....DOS file specification.

Required: ..... Yes, Opus will not run without it.

Example: ..... USES SCHEDULE c:\opus\sched.dat

**VIDEO <kind>**

Definition: .....Specifies which video mode Opus will use at the sysop's local console.

Kind	Exmple	Meaning
DOS	VIDEO DOS	DOS is the default mode and will work with most IBM compatible systems.
FOSSIL	VIDEO FOSSIL	FOSSIL is slower, but should work on just about any system that can run MS-DOS.
IBM	VIDEO IBM	IBM mode is very fast because it writes directly to video RAM; it will therefore only work on true IBM compatibles. (80X25 only)

**TECHNICAL NOTES:**

ANSI screens will not show correctly on the local console, if IBM mode is enabled. Avatar screen codes will show correctly. It may therefore be worthwhile converting all your ANSI video codes to Avatar video codes.

If you are using a multitasker, and you find your Opus partition(s) or window(s) bleed through into your other partition(s) or window(s), or disappear, do not use the IBM video mode option.

If you wish to be able to CONTROL-C from the local keyboard during bulletins etc, do not use the IBM video mode option.

Required: .....No Default of VIDEO DOS will be used.

**2.8.2 LANGUAGE SECTION**

Definition:.....Specifies the beginning of the Language section of the control file.

Required: .....Yes

**CHARSET <filename>**

Definition: .....Opus V1.70 supports the alternate character sets as set out in FSC-0050 and OPMED 3.0. This option specifies where Opus can locate the translation table file (see further CHARSET in the Message Area Definition Section below).

Range:.....DOS file specification

Required: .....No

Example: .....CHARSET c:\opus\iso\opmed.tlt

**END LANGUAGE SECTION**

Definition: .....Specifies end of the language section of the control file.

Required: .....Yes

**LANGDIR <path>**

Definition: .....Specifies the location of the language files (.SYL and .USL)

Range: .....DOS path specification.

Required: ..... Yes

Example: .....LANGDIR c:\opus\language\

**LANGUAGE <filename> <string>**

Definition: .....Specifies the Opus V1.70 language filename and how the language choice should be displayed to a user. The name of the file is the first 8 characters up to the first space. Opus will display everything after the first space, or the filename if it does not find any space characters. This is also the name that will show up when new users call and on the History menu for the A)lter language command.

Range: .....<filename> Any eight-character language name. <string> Up to 50 printable ASCII characters.

You may specify up to twelve languages. The first one listed in the control file is language number 1, the second is language number 2, and so on.

Required: .....Yes, at least one language <filename> must be specified; the descriptive <string> is optional.

Examples: .....LANGUAGE ENGLISH English English LANGUAGE AMERICA 'Merican English  
LANGUAGE FRENCH French

The above would appear on the History menu (and would give a new user a menu) like this:

```
User Language: 1) English English 2) 'Merican English 3) French
Select: _
```

**NOHIGH**

Definition: .....Specifies that Opus V1.70 will change the default high bit line-drawing characters to their ASCII equivalents. Opus will also require exact case matches for strings such as user logon names, passwords, file names, messages To: people.

Required: .....No

**SYSOP <number>**

Definition: .....Specifies the language selection for the sysop's local console

Range: ..... 1-12

Required: ..... Yes

Example: .....SYSOP 1

If the language number 1 is defined to be French, then the sysop would see French on his local console provided the sysop has the relevant French .SYL and .USL files in the language directory (see LANGDIR).



**USER <number>**

Definition: ..... Specifies default language for a new user.

Range: ..... 1-12

Required: ..... Yes

Example: ..... USER 1

If the language number 1 is defined to be French, then the user would see French when s/he first logs on. The user can select a new language at the Change menu, if more than one language is available and the user has access to the Alter Language command.

**2.8.3 EQUIPMENT SECTION**

Definition: ..... Specifies the beginning of the equipment section of the control file.

Required: ..... Yes

**AUTOANSWER <string>**

Definition: ..... Specifies that Opus V1.70 should detect the modem RING result code and send an "answer" string to the modem to cause the it to go off-hook and start the answer sequence. If you use this option, make sure that your modem's S0 register is set to 0 so that the modem will not answer incoming calls.

Range: ..... Up to 50 printable ASCII characters with same range and meaning as the MODEM INIT string (see above).

Required: ..... No (provided the modem is set to autoanswer by means of the modem S0 register), but recommended. If Opus should crash for whatever reason, the modem will just allow calls from users and other mailers to ring out. If you were instead relying on the modem to answer the calls, the calls would be answered regardless and users and other sysops could waste long distance (or even international) phone charges.

Examples: ..... ATA| ~A~T~A~|

**BAUD CANSTEADY**

Definition: ..... Specifies that Opus V1.70 will lock the baud rate on ARQ connects and let it float on non-ARQ connects. For sysops using USR 14.4K and Dual Standard modems which support the &B2 switch. Consult your HST manual for more information on &B2.

Required: ..... Heckiflknow <TM Doug Boone>. If you are having problems with a program passing a low baud rate to Opus, and Opus resets the baud rate at its max baud rate, and you have a USR modem with &B2 enabled, then try this option.

Range:.....300, 1200, 2400, 4800, 9600, 19200, 38400

Required: ..... Yes

Examples:.....BAUD MAXIMUM 2400 BAUD MAXIMUM 19200

Definition: .....Specifies whether the port baud rate will be locked at a given speed. Locking at a higher speed such as 19200 or 38400 will allow higher throughput speeds with certain high speed modems or modems using MNP compression, if your computer can handle the communications overhead for the higher speeds.

Required:.....No. Not recommended for 8088 or 8086 processor based computers. Even 80286/80386/80486 computers may require an NS16550AFN UART be substituted for the usual NS16450 or 8250 UART to keep up with high speeds.

This parameter is no longer required for use with most of the newer fossil drivers. Lock the port with the fossil driver instead. Locking the fossil directly is preferred because it will allow most door programs to work.

Definition: ..... Specifies the string that is sent to the modem when you are accessing Opus from the local keyboard or when Opus exits to a batch file. As a minimum, you should disable the modem from answering, to prevent long distance callers from connecting to an inaccessible Opus. If your modem-phone combination can do it, you should also send the string to take the phone off-hook, giving a busy signal. This will allow a caller or other mailer program to quickly detect that your system is unavailable if the calling modem supports call progress monitoring.

Range:.....Up to 50 printable ASCII characters with same range and meaning as the MODEM  
INIT string (see above).

Required: .....No, but highly recommended.

Examples:.....v Drop Data Terminal Ready signal to prevent answering. Works with all most all modems.

ATS0=0H| Defeats auto-answer capability of modem and takes phone off-hook.



**DIAL PREDIAL1 <string>****DIAL PREDIAL2 <string>****DIAL PREDIAL3 <string>****DIAL PREDIAL4 <string>**

Definition: ..... Specifies different setups for dialing outbound matrix calls. Opus V1.70 will check the Modem byte in NODELIST.DAT, and if it is set to one of these values Opus will use the Predial(n) that corresponds to the first match. According to Parselst 1.30+, it recognizes two modem types, HST and PEP (HST = 1 and PEP = 2). If no modem flag has been set for this node, Opus will use its default Dial Prefix string.

Range: ..... Up to 50 printable ASCII characters with the same range and meaning as the MODEM INIT string (see INIT).

Required: ..... No

Example: ..... DIAL PREDIAL1 ATB0DT

**DIAL PREFIX <string>**

Definition: ..... Specifies the default dial string sent to the modem before the telephone number when Opus attempts to make an outbound Matrix call.

Range: ..... Up to 50 printable ASCII characters with the same range and meaning as Modem Init string (see above). Note that a suffix is not specified with this string. You can also place global information that applies to all outbound calls here. For example, if your telephone switchboard requires that you dial a '9' for accessing an outside line, you could put it here in the DIAL PREFIX.

Required: ..... Yes

Examples: ..... ATDT ATDP9,

**DIAL SUFFIX <string>**

Definition: ..... Specifies the dial string sent to the modem after the telephone number to complete the dialing string for an outbound Matrix call. Note: most modems require a carriage return.

Range: ..... Up to 50 printable ASCII characters with same range and meaning as the Modem Init string (see INIT).

Required: ..... Yes

Example: ..... |

**END EQUIPMENT SECTION**

Definition: ..... Specifies the end of the equipment section in the control file.

Required: ..... Yes

**INIT <string>**

Definition: .....Specifies the string sent to the modem on startup and after each user and matrix session.

NOTE: Opus also sends the init string each time it checks the outbound directory, provided it has been set up to send mail. This is done to help keep some finicky modems "alive". If you see your modem lights flash occasionally when Opus is at the ready prompt, it is this "tickling" activity that is probably the cause.

Range: .....Up to 50 printable ASCII characters with some that have special meaning:

- v (lowercase V) lower DTR
- ^ (caret or circumflex) raise DTR
- | (concatenate) carriage return
- ~ (tilde) slight pause
- b send break

Other Hayes compatible codes that may be important:

- Q0 (\*) result messages are sent
- E0 (\*) commands sent to the modem are not echoed
- H0 hang up (if off-hook for any reason)
- M0 speaker always off
- V1 (\*) send verbal responses, NOT NUMERIC
- S0=1 answer a call on the first ring
- X4 use EXTENDED verbal responses
- &W write string to modem's NVRAM (required for the Compucom Speedmodem only)

(\*) required for Opus

Required: ..... Yes

Example: .....ATQ0E0M0V1S0=1X4|

**MASK CARRIER <number>**

Definition: .....Specifies the bit mask used for carrier detect.

Range: .....1, 2, 4, 8, 16, 32, 64, 128

Required: .....No (defaults to 128 which is used by most Hayes compatible modems)

Example: .....MASK CARRIER 128

**MASK HANDSHAKING <kind>**

Definition: .....Specifies hardware handshaking, the way for your computer and your modem to keep in sync with each other. It is also used between remote systems.

Range:.....Any combination of XON and either CTS or DSR. If you need to have more than one type, put several handshaking declarations in the control file. Do not put more than one declaration on a single line.

Required:.....Most modems that run faster than 2400 baud or use MNP compression require CTS hardware handshaking.

IMPORTANT: This mask also affects the user's ability to use control-S and control-C. If you have a modem that requires CTS or DSR, it would be a good idea to use both CTS (or DSR) \*and\* XON.

For high-speed modems (faster than 2400 baud) you MUST use CTS and XON. In addition, your FOSSIL program needs to support CTS/RTS.

Examples:.....MASK HANDSHAKING CTS  
MASK HANDSHAKING DSR  
MASK HANDSHAKING XON

**MOUSE**

Definition: .....Specifies that Opus V1.70 should enable mouse support. Opus only knows about button presses being equivalent to key presses. The left button is a carriage return and the right button is a "no" character.

Required:.....Only by Mickey and avid rodent sysops

**OUTPUT <kind>**

Definition: .....Specifies the default port or local keyboard mode that Opus will use when invoked. This option can be overridden by command line parameters (see command line parameters section).

Range:.....COM1-COM16 or Local. You can only specify one of these.

Required:.....Yes

Examples:.....OUTPUT COM2  
OUTPUT COM1  
OUTPUT LOCAL

**SEND BREAK TO CLEAR BUFFER**

Definition: .....Causes Opus to send a break signal if it needs to clear the outbound transmit buffer for some reason.

Required:.....Yes, if you use a U.S. Robotics HST modem.

NOTE: IF YOU USE A U.S. ROBOTICS HST MODEM, SET THE BREAK PARAMETER TO

&Y0

WHICH MEANS DUMP THE OUTBOUND BUFFER, BUT DO NOT FORWARD THE BREAK TO THE OTHER SYSTEM. THIS IS NOT THE FACTORY DEFAULT.

**SLOW MODEM**

Definition: ..... Specifies that Opus should insert a pause between each character sent to the modem in the initialization, busy, dial, predial and answer strings.

Required: ..... No. If you suspect that modem strings are not being handled properly by your modem, this option may possibly help the situation.

**2.8.4 MATRIX AND ECHOMAIL SECTION**

Definition: ..... Specifies the beginning of the parameters section for matrix and echomail.

Required: ..... Yes

**ADDRESS <zone:net/node.point>**

Definition: ..... Specifies the full FidoNet network address for your Opus-CBCS.

Range: ..... Zone:net/node.point You may have up to fifteen addresses. Use a point number of 0 as the default for a full node address.

Whereas in Opus V1.1x, the first address listed was considered your primary address, Opus V1.70 takes a new approach. If you have multiple addresses Opus will also its "best fit" address in the YooHoo packet instead of the first address. Only if there is no better match will Opus use your first listed address.

Required: ..... Yes

Example: ..... Address 1:119/5.0  
 Address 1:119/25.0  
 Address 99:103/5.0  
 Address 99:103/0.0  
 Address 99:123/0.0

<u>If a call comes from:</u>	<u>Opus will answer as:</u>
outside of zone 99	1:119/5.0
99:103/*	99:103/5.0
99:123/*	99:123/0.0
99:(any other net)/*	99:103/5.0

**AFTER ARCMail EXIT <number>**

Definition: ..... Specifies that Opus should exit after receipt of compressed mail with the specified DOS errorlevel. Use this option if you wish to do all compressed mail processing external to Opus.

Range: ..... DOS errorlevel specification.

Required: ..... No

Example: ..... AFTER ARCMail EXIT 14



## **AFTER ARCMail EXTRACT PACKETS**

Definition: .....Specifies that Opus will automatically extract packets from incoming arcmail on receipt. Be sure you have adequate memory available to do this and that the appropriate decompression utility to do the extraction is in the DOS path.

Opus will identify incoming mail that has been packed as ARCMail, ARJmail, ZIPmail, ZOOmail, and LZHmail and will use the correct program to unpack the mail archive \*if\* it finds the program in the environment path.

The default programs are: ARCE.COM, ARJ.EXE, PKUNZIP.EXE, ZOO.EXE and LHA.EXE.

Required: .....No

## **AFTER EDIT EXIT <number>**

Definition: .....Specifies that Opus should exit with the specified DOS errorlevel after a message is entered in the matrix/netmail area and the caller has logged off. This errorlevel would be used for the automatic processing of the message(s) the user entered in the matrix/netmail area by an external mail packer.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER EDIT EXIT 15

## **AFTER INBOUND EXIT <number>**

Definition: .....Specifies that Opus should exit after receiving mail with the specified DOS errorlevel. You would use this if you wished to use an external mail processor for unpacking and tossing/scanning mail.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER INBOUND EXIT 12

## **AFTER INBOUND TOSS ECHOMAIL**

Definition: .....Specifies that Opus will unpack and toss inbound mail. This option will override the AFTER INBOUND EXIT option.

Required: .....Yes, if you want to have Opus automatically unpack and toss inbound mail.

### **AFTER LOCAL EXIT <number>**

Definition: .....Specifies that Opus V1.70 should exit with the specified DOS errorlevel after a message is entered in a local (ie, non-matrix and non-echomail) message area and the caller has logged off. Useful for people running mail waiting programs that compile a new list of messages after each caller.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER LOCAL EXIT 16

### **AFTER MAIL EXIT <number>**

Definition: .....Specifies that Opus V1.70 should exit after any mail exchange with the specified DOS errorlevel. This is primarily for use by people who are experimenting with non-FTSC mail to convert their mail to FTSC mail or whatever (eg, decrypt your mail).

Note:

It will only be used if Opus is doing mail internally (ie Opus is your mailer).

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER MAIL 14

### **AFTER TOSS EXIT <number>**

Definition: .....Specifies that Opus should exit after tossing mail with the specified DOS errorlevel. You would use this if you wished to use an external mail processor for scanning or other processing.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER TOSS EXIT 13

### **BOSSNODE <net/node>**

Definition: .....Specifies that your Opus V1.70 is being run as a Point system. Your primary network address is Zone:PointNet/Node.

Opus cannot be run as both a boss node AND a point system; the two options are mutually exclusive.

Range: .....The FidoNet address of your boss node.

Required: .....No

Example: .....BOSSNODE 119/5.0



## CRASHES <Number>

Definition: .....Sets the number of times Opus will try to make a CRASH call in a row. Normal \*.OUT, \*.DUT, \*.FLO and \*.DLO calls are each made once in a rotation, but CRASH calls are dialed out repeatedly for a fixed number of tries before moving on to the next outbound call.

Range: ..... 1 - 255. Default is 10.

Required: .....No, but if users are having trouble logging on because of outbound mail calls, you might want to adjust this to a lower value.

Example: .....CRASHES 5

## ECHO ECONOSCAN

Definition: When you use this command with the built-in Echomail handling services, Opus will completely toss and scan one bundle at a time. Normally, all bundles are decompressed and tossed with scanning following as the last procedure.

Under EconoScan, Opus only tosses one packet at a time.

THIS IS ONLY USEFUL FOR THOSE WHO HAVE NUMEROUS "PASSTHROUGH" Echomail areas on a small storage device like a RAM disk.

Using EconoScan, Echomail processing will take longer... but presumably it will only be used by those who are using RAM disks.

This assumes you will have lots of small .PKT files. It will not help in cases where you have one 300 meg bundle to toss and scan.

Required: No

## ECHO GUARD

Definition: Specifies that Opus should only toss echomail that is received from a session-level passworded connection. Echomail that comes from a non-passworded connection will be logged and orphaned.

Required: No

## ECHO MSGID

Definition: .....Specifies that Opus will generate MSGID codes, in addition to EID codes in echomail messages originated by your Opus. MSGID codes are an identification code stored in the message that may be useful to some mail processing software on other systems.

Required: .....No. You should only enable this if you are aware of a specific need on other systems to have a MSGID implanted in your messages.

**ECHO NOALIAS**

Definition: .....Specifies that Opus V1.70 is not to put any of your alias network addresses in echomail seen-bys in echomail message areas. Only the network address specified in the echomail message area definition will be used, all others (even in the same zone) will be ignored.

Required: .....No

**END MATRIX AND ECHOMAIL SECTION**

Definition: .....Specifies the end of the matrix and echomail parameters section in the Opus control file.

Required: .....Yes

**EXTMAILER <string> <exit>**

Definition: .....Specifies that Opus V1.70 should look for the string you specify and, if it is given as a name or while Opus is in MAIL-ONLY mode, Opus will exit with the specified DOS errorlevel plus the current errorlevel offset (if any).

If <string> calls, Opus will create "MAILER.BAT" and specify:

- the current baud rate;
- the current port number (1=COM1:, 2=COM2: etc);
- the number of minutes until the next event;
- the exit errorlevel,
- and then exit with the specified <exit> errorlevel.

A sample MAILER.BAT created by Opus V1.70 might contain this:

```
EXTMAIL 9600 2 68 100
```

EXTMAIL is a batch file which you should create. You can then make use of the parameters which Opus has passed in the MAILER.BAT file. For example, your EXTMAIL.BAT might contain the following:

```
MAILER.EXE -b%1 -p%2 -t%3 NERF
```

Your mailer program, MAILER.EXE would then be called with the command line -b9600 -p2 -t68 and when it exits it should invoke your Opus startup batch file (NERF.BAT) and restart Opus.

Range: .....<string> Up to 50 printable ASCII characters

Note: it is case sensitive. <exit> DOS errorlevel specification.

Required: .....No

Example: .....EXTMAILER Gimme UUCP 99

### **GUUCP <Zone:Net/Node.Point>**

Definition: .....Specifies the FULL FidoNet address of the closest/least expensive UUCP gateway. Opus V1.70 will automatically route messages to a UUCP address through this gateway. Who's a gateway? Look in the FidoNet nodelist for systems which have the "GUUCP" flag listed.

Range: .....The full FidoNet address of the gateway system

Required: .....No

Example: .....GUUCP 3:711/501.0

### **LEAVE ECHOTOSS**

Definition: .....Specifies that Opus V1.70 should not delete the ECHOTOSS.LOG after it has scanned the areas listed in that file.

Required: .....No

### **LOG ECHOMAIL**

Definition: .....Specifies that Opus should log each echo area to a file named ECHOTOSS.LOG when it tosses mail in that area or an online user creates a new message in that area.

Required: No

### **MAILER COMMAND <command line>**

Definition: .....Specifies the command line that will be used to load an external mailer over Opus. Memory intense, but Opus loads and initializes, and then goes resident and loads the mailer. When the mailer recognizes a human and drops out, Opus is already in memory and ready to go.

Range: .....DOS command line.

Required: .....Yes, if external mailer is specified.

Example: .....MAILER COMMAND c:\bink\bt.exe share

### **MAILER LOAD <number>**

Definition: .....Specifies whether Opus will use its own internal mailer or an external mailer as the default mailer. You can override this from the event file by declaring a mailer event.

Range: .....0 = Use Opus internal mailer  
1 = Use external mailer specified by the MAILER COMMAND (see above)  
2 = Call external mailer specified by the MAILER COMMAND (see above)

Required: .....No, defaults to Opus internal mailer

Example: .....MAILER LOAD 1

**MAXECHO <number>**

Definition: .....Specifies the maximum kilobyte size of messages that Opus V1.70 should scan out when handling echomail. Messages larger than the specified size will simply be skipped. Opus 1.1x imposed a 10K limit.

NOTE: .....If you are running on a multitasking system, the size of this number MAY have an effect. If your Opus windows/partitions are small and you pick 60 as the size of echomail messages to handle, you may get errors when trying to process echomail because Opus cannot find any unused far memory.

Range: .....3-60 (Default is 10)

Required: .....No

Example: .....MAXECHO 12

**MESSAGE EDIT ASK CRASH <priv>**

Definition: .....Specifies that Opus will ask a user of this privilege level or higher if they wish to create a message marked CRASH in the matrix/netmail area. This attribute will cause the message to go out without any cost restriction and should be granted with great care.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASK CRASH Asstsysop

**MESSAGE EDIT ASK FILEATTACH <priv>**

Definition: .....Specifies that Opus will ask a user of this privilege level or higher if they wish to create a message with a file attach in the matrix/netmail area. This attribute could cause security violations of your BBS and should be granted with care.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASK FILEATTACH Sysop

**MESSAGE EDIT ASK FROMFILE <priv>**

OBSOLETE. (In Opus V1.70, see the ED\_DISK option for editor menus in section 3 of this manual.)

**MESSAGE EDIT ASK HOLD <priv>**

Definition: .....Specifies that Opus will ask a user of this privilege level or higher if they wish to create a message marked HOLD in the matrix/netmail area.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASK HOLD Clerk

### **MESSAGE EDIT ASK KILLSSENT <priv>**

Definition: ..... Specifies that Opus will ask a user of this privilege level or higher if they wish to create a message that will be deleted after it is sent in the matrix/netmail area.

Range: ..... Opus privilege level.

Required: ..... No

Example: ..... MESSAGE EDIT ASK KILLSSENT Clerk

### **MESSAGE EDIT ASK PRIVATE <priv>**

Definition: ..... Specifies that Opus will ask a user of this privilege level or higher if they wish to create a private message when entering a message in a message area which will accept private messages.

Range: ..... Opus privilege level.

Required: ..... No

Example: ..... MESSAGE EDIT ASK PRIVATE Privil

### **MESSAGE EDIT ASK REQUEST**

Definition: ..... Specifies that Opus V1.70 will ask the user whether to set the file REQUEST attribute when a user of this privilege level or higher creates a message in the matrix/netmail area.

Range: ..... Opus privilege level name.

Required: ..... No

Example: ..... MESSAGE EDIT ASK REQUEST Worthy

### **MESSAGE EDIT ASK UPDATE <priv>**

Definition: ..... Specifies that Opus V1.70 will ask the user whether to set file request UPDATE attribute when a user of this privilege level or higher creates a message in the matrix/netmail area.

Range: ..... Opus privilege level name.

Required: ..... No

Example: ..... MESSAGE EDIT ASK UPDATE Worthy



**MESSAGE EDIT ASK UUCP <priv>**

Definition: .....Specifies that Opus V1.70 will ask a user of this privilege level or higher who is replying to a UUCP message in the matrix/netmail message area whether to delete the UUCP address that Opus will otherwise automatically read out of the original message. If the user does, then a new UUCP address will need to be typed in as the first line of the reply. This option will also appear on the Message Handling menu.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASK UUCP Normal

**MESSAGE EDIT ASSUME CRASH <priv>**

Definition: .....Specifies that Opus will assume a CRASH attribute when a user of this privilege level or higher creates a message in the matrix/netmail area. This attribute should be used with caution as it may cause your system to make long distance phone calls to deliver mail.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASSUME CRASH Sysop

**MESSAGE EDIT ASSUME HOLD <priv>**

Definition: .....Specifies that Opus will assume a HOLD attribute when a user of this privilege level or higher creates a message in the matrix/netmail area.

Range: .....Opus privilege level name.

Required: .....No

Example: .....MESSAGE EDIT ASSUME HOLD Worthy

**MESSAGE EDIT ASSUME KILLSSENT <priv>**

Definition: .....Specifies that Opus V1.70 will assume a killsent (delete after sending) attribute when a user of this privilege level or higher creates a message in the matrix/netmail area.

Doug found that Opus versions before V1.70 had always worked so that a user's privilege level had to be BELOW the ASSUME level for the ASK level, so that if: MESSAGE EDIT ASSUME KILLSSENT Twit MESSAGE EDIT ASK KILLSSENT Sysop no one would ever be asked about KillSent. Opus would just turn it on for everyone. Now Opus will respect this and will ask Sysop privilege people whether or not they want messages to be killed after sending them.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASSUME KILLSSENT Normal

### **MESSAGE EDIT ASSUME PRIVATE <priv>**

Definition: .....Specifies that Opus will assume a private attribute when a user of this privilege level or higher creates a message in the matrix/netmail area.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE EDIT ASSUME PRIVATE Disgrace

### **MESSAGE SHOW CTL\_A TO <priv>**

Definition: .....Specifies that Opus will display any lines beginning with a control-A (x01) character to a user with this privilege or higher. This option would normally be set to assistant sysop level or higher, because the control-A lines have no meaning to the average user. Control-A's are used in Echomail messages as a "command" delimiter. Such things as the area's name usually follow's a control-A character.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE SHOW CTL\_A TO Sysop

### **MESSAGE SHOW SEENBY TO <priv>**

Definition: .....Specifies that Opus will display SEEN-BY lines in echomail message areas to a user with this privilege level or higher. This option would normally be set to assistant sysop level or higher because the SEEN-BY lines have no meaning to the average user.

Range: .....Opus privilege level.

Required: .....No

Example: .....MESSAGE SHOW SEEN-BY TO Sysop

### **MINIMUM BAUD <number>**

Definition: Specifies the minimum connect speed in BPS at which Opus will conduct a mail session.

Range: 300-38400 BPS

Required: No

Example: MINIMUM BAUD 1200

### **PATH BADECHO <path>**

Definition: .....Specifies where Opus will toss "bad" mail messages when processing mail. The "bad" messages are ones that violate security for intransit mail or echomail, or are bound for an unrecognized area.

Range: .....DOS path specification.

Required: .....No

Example: .....PATH BADECHO c:\opus\messages\bad\

### **PATH INBOUND FILES <path>**

Definition: .....Specifies the path for your inbound matrix packets and files.

Range: .....DOS Path specification.

Required: .....Yes

Example: .....PATH INBOUND FILES c:\opus\files\net\

### **PATH INBOUND MESSAGES <path>**

Definition: .....Specifies the path for your matrix/netmail area. If you wish to access the messages in this area from Opus, you should specify a message area which points to this directory. Normally the area should also have the MATRIX attribute set.

Range: .....DOS path specification.

Required: .....Yes

Example: .....PATH INBOUND MESSAGES c:\opus\messages\net\

### **PATH INMAIL <path>**

OBSOLETE (used in Opus 1.1x). See the new STATUS\_DIR control option above.

### **PATH NETINFO <path>**

Definition: .....Specifies the DOS path to where your nodelist files will be kept.

Range: .....DOS path specification.

Required: .....Yes

Example: .....PATH NETINFO c:\opus\nodelist\

**PATH OUTBOUND HOLDAREA <path\>**

Definition: .....Specifies the path to where your zone's outbound mail will be stored.

Opus V1.70 supports multiple outbound zone hold areas as supported by Binkley and Qmail. Note, however, that the zoned outbound directories **MUST BE IN NUMERICAL ORDER** in your directory structure (ie, outbound.008 must occur after outbound.007 which must occur after outbound.006 and so on). Watch out for disk defragmenters changing the order!

Range: .....DOS path specification without an extension. Other zone's outbound hold areas take the form of your zone's outbound directory area plus an hexadecimal zone number extension.

Required: ..... Yes

Example: .....PATH OUTBOUND HOLDAREA c:\opus\outbound\

Assuming you were in Zone 3, if you wished to send mail to zones 1, 2, 4, 5, 6 and 16 you would then need to make the following additional directories on your hard disk in numerical order after your c:\opus\outbound:

c:\opus\outbound.001  
c:\opus\outbound.002  
c:\opus\outbound.004  
c:\opus\outbound.005  
c:\opus\outbound.006  
c:\opus\outbound.00F

**POINTNET <number>**

Definition: .....Specifies that Opus V1.70 is running as a "boss" node for a PointNet. Opus will strip off the point's SEEN- BY and Origin and replace it with your address plus their point number.

Opus will not save the SEEN-BYs for Points in the messages. If you delete l.msg and rescan an area, your echomail processor will send out duplicates to all the Points in your PointNet.

Opus cannot be run as both a boss node AND a point system; the two options are mutually exclusive.

Range: .....0-32767

Required: .....No

Example: .....POINTNET 1001

**REFUSE ALL INTRANSIT**

Definition: .....Specifies that Opus will not process or send any intransit (routed) mail. This feature is for security to protect you from someone sending mail addressed to another system via your system on your nickel. For example, if your system is not correctly setup, it may try to deliver the mail to another system which is a long distance phone call for you. Systems have been known to make INTERNATIONAL phone calls in these situations!

Required: .....No, but highly recommended, unless you know what you are doing. If you are a HUB or HOST system, DO NOT use this option.

### **REFUSE ATTRIBUTE INTRANSIT**

Definition: .....Specifies that Opus will not process intransit mail that has a Crash, Request, or File Attach attribute.

Required: .....No

### **REFUSE EXITS**

Definition: .....Specifies that Opus V1.70 will not exit via a DOS errorlevel after mail sessions regardless of any exits specified in the Opus Control File. This is equivalent to the "suppress exits" option in the Opus event schedule.

Required: .....No

### **REFUSE FILE REQUESTS**

Definition: .....Specifies that Opus should not accept file requests. This behaviour may be modified by behaviour windows (see events section).

Required: .....No

### **REFUSE HUMAN CALLERS**

Definition: .....Specifies that Opus should not accept human callers. This behaviour may be modified by behaviour windows (see events section).

Required: .....No

### **REFUSE INBOUND TRAFFIC**

Definition: .....Specifies that Opus will not accept any inbound matrix/netmail traffic. This behaviour may be modified by behaviour windows (see events section).

Required: .....No. This setting should NOT be used by HOSTS or HUBS or other FidoNet Coordinators.

### **REFUSE NONLOCAL INTRANSIT**

Definition: .....Specifies that Opus will process intransit mail that is destined for local nodes only. This allows you to route mail without concern for sending mail that goes to non-local calls. This option might be useful for an inbound net host.

Required: .....No

### **REFUSE UNPASSWORDED INTRANSIT**

Definition: .....Specifies that Opus will only process intransit mail that was received from a matrix session that has a session-level password. This is useful for allowing selected nodes to route mail through your system.

Required: .....No



## SCAN ECHOMAIL

Definition: ..... Specifies that the Opus internal mail scanner will be used for processing echomail after a user has entered mail in an echo area, or messages have been tossed by Opus into an echo area.

Required: No

## SEND CM

Definition: ..... Specifies that Opus will send mail to nodes that accept continuous mail only (as defined in the nodelist). This behaviour may be modified by behaviour windows (see events section).

### REMEMBER:

You do not mark mail as crash to get it to a node that will accept continuous mail. Opus looks in the nodelist to find out if the node is marked with the CM flag and then sends it. This is a change from Opus version 1.0x. If you mark mail as crash, Opus treats it as SEND IT NOW and will make a concerted effort to get the mail through disregarding cost.

Required: .....No

## SEND HOST

Definition: ..... Specifies that when Opus V1.70 cannot find an address, it should to send the mail to the Net Host instead of marking it as "undialable".

NOTE: ..... This only works with the version 7 nodelist files. OPARSE.EXE will accept a keyword in the CFG file of "HOST" which tells it to only include the Zone administrative nodes, the Net and Region Hosts, and your Net. All the other systems will be left out of the Opus nodelist file, but when this feature is combined with the "SEND HOST" command, all the other systems are still reachable by host routing.

Required: .....No

## SEND LOCAL

Definition: ..... Specifies that Opus will only send mail that has no cost associated with it. This behaviour may be modified by behaviour windows (see events section).

Required: .....No

## SEND NOTHING

Definition: ..... Specifies that Opus will not send any mail. This behaviour may be modified by behaviour windows (see events section).

Required: .....No

**USES ECHOTOSS <filename>**

Definition: .....Specifies the filename for Opus V1.70 to find/put the ECHOTOSS.LOG (you can now change the name, but be aware that many programs look for the filename ECHOTOSS.LOG). Opus V1.70 uses this log file for its internal echomail scanner, so external programs that use the log file will be picked up by Opus. Note: the LOG ECHOMAIL or SCAN ECHOMAIL options must also be enabled for this file to be created.

If you force Opus to do a scan, only areas that are listed in log will be scanned if the log file exists, otherwise Opus will scan ALL areas. The Scan event still scans all areas whether the log file exists or not.

FIXTOSS.EXE is a program that can be used to eliminate duplicate area entries from ECHOTOSS.LOG for programs that are not clever enough to ignore duplicate listings.

Range: .....DOS filename specification.

Required: .....No

Example: .....USES ECHOTOSS c:\opus\logs\echotoss.log

**USES FILEREQ ABOUT <filename>**

Definition: .....Specifies the name of a file that will be sent when a file request fails or when a file request for "ABOUT" is received. This file should contain information about your BBS. For example, the types of files you carry, the magic filenames you have available for request, the times when your system will accept file requests, any file request restrictions, etc. and the purpose of your system.

Range: .....DOS file specification.

Required: .....No

Example: .....USES FILEREQ ABOUT c:\opus\fireside.zip

**USES FILEREQ FILELIST <filename>**

Definition: .....Specifies the name of the file that will be sent to the system requesting a file by the name of "FILES". This file should contain a list of files and their descriptions that are available for file request. It is suggested that this file be named as a derivative of your address such as xxxxyyyy.zip, where xxxx is your net number and yyyy is your node number. As a courtesy to long distance and international callers, you should compress the file using a popular file compressor. This helps keep their phone costs down when requesting a list of files from your system.

Range: .....DOS file specification.

Required: .....No

Example: .....USES FILEREQ FILELIST c:\opus\01060114.zip

**USES FILEREQ OKFILE <filename>**

Definition: .....Specifies the name of the file that tells Opus what is available when a file request comes in from another system.

Range: .....DOS file specification.

Required: .....No

Example: .....USES FILEREQ OKFILE c:\opus\okfile.lst

An example OKFILE is shown here. (The use of the OKFILE is explained in detail in the Opus Operations manual).

```
@echolist f:\fidonet\echo116.lst
@Opusbeta !pword1 f:\opusbeta\*. *
f:\communic\*. *
f:\games\*. *
f:\desqview\*. *
f:\net\*. * !pword2
f:\upload\*. * !pword3
$anything !pword4 anything.bat %d %d
$fromwork !pword5 fromwork.bat %d %d
$getxpres !pword6 getxpres.bat %04x %04x
$putxpres !pword7 putxpres.bat %04x %04x
```

**USES UUCPLIST <filename>**

Definition: .....Specifies that Opus V1.70 should use a UUCPlist. UUCPlist is like FidoUser.Lst, a list of names and addresses. The difference is that UUCPlist is UUCP addresses, not FidoNet addresses. If this option is specified, you simply enter Netmail messages with the person's name on the address line and Opus will look them up in UUCPlist first, then either FidoUser (version 6 nodelist) or SYSOP.NDX (version 7 nodelist).

Range: .....DOS file specification.

Required: .....No

Example: .....USES UUCPLIST c:\opus\uucp\names.lst

The format of the file is as follows:

```
=====
Vincent Cate vac@cs.cmu.edu
Vincent Investors space-investors-request@cs.cmu.edu
Chuck Divine xrcjd@scfvm.gsfc.nasa.gov
Joseph Davis nfjad@alaska.bitnet
Patrick Heopfner heopfner@heasfs.gsfc.nasa.gov
Bill Higgins higgins@fnal.bitnet
T.S. Kelso tkelso@blackbird.afit.af.mil
Short Kelso tkelso@afit.af.mil
Cary Oler oler@hg.uleth.ca
Tihamer Toth-Fejel ttf@iti.org
Bev Freed freed@nss.fidonet.org
=====
```

So if you wanted to enter a NetMail message to Bev Freed that would go through the UUCP gateway, all you would have to do is enter the message header and give "Bev Freed" as the address.

Opus would look up "Bev Freed" in UUCPlist and would store the UUCP address on the top line of the body of the message, (when it is saved), put "UUCP" on the To: line, and the UUCP gateway for the FidoNet address.

You might notice that there's a "Vincent Cate" and a "Vincent Investors". This is because this person has at least two different addresses so there is a different name for the separate addressing scheme.

## USES VERSION6

Definition: .....Specifies that your Opus system is still using the "old" version 6 nodelist files instead of the new Opus V1.70 version 7 nodelist files.

Opus V1.70 defaults to the advanced version 7 nodelist; you must specify this option if you wish to continue to use the old version 6 nodelist files.

Version 7 nodelist files have several important advantages:

The Version 7 nodelist is installed as the DEFAULT nodelist for Opus V1.70. You can over-ride this by putting VERSION6 in the Matrix Section of your control file.

What does the Version 7 nodelist give you?

- 40% of your disk space back;
- sysop names when doing lists of boards;
- automatic insertion of Sysop name when you type in an address;
- ability to poll from the keyboard using sysop names without having to have the huge FIDUSER.LST available;
- more flexibility when building the nodelist;
- protection from Guido's "progress" committee.

To create the version 7 nodelist files you need to run OPARSE.EXE.

Required: .....Opus V1.70 defaults to the new, more advanced, Guido approved version 7 nodelist files; you MUST specify this option IF you wish to continue to use the old, bulky version 6 nodelist files.

NOTE: this option must NOT be specified if you wish to take advantage of the SEND HOST option (see below).

## 2.8.5 SESSION SECTION

Definition: .....Specifies the beginning of the session section of the control file.

Required: .....Yes

### AFTER CALL EXIT <number>

Definition: .....Specifies the DOS errorlevel which Opus should exit with after each human caller logs off.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER CALL EXIT 5

### AFTER Reelog Exit <number>

Definition: .....Defines the exit errorlevel to use after a relogged user is disconnected so that Opus can be cycled up to the normal command line in the batch file.

Required: .....Only if you use the '-o' option on the command line.

Here is an example of a batch file for Opus that uses the RELOG feature with Doorway assuming that if a user chooses Doorway from a menu and it is a EXIT 120.

```
:loop ←
opus bbs %1 %2
if errorlevel 120 goto outside ←
rem
rem Put other errorlevel tests here
rem
if errorlevel 1 goto end
goto loop
:outside ←
DOORWAY COM2 /S:* /G:ON /K:600 /V:D^U /M:900 /B:Z /O:T /C:DOS
goto relog ←
:RELOG ←
rem
rem This is where we allow users to get back into
rem Opus without logging on
rem Notice the "-o" on the command line.
rem
opus bbs -o
if errorlevel 120 goto outside:
rem
rem Put other errorlevel tests here
rem
rem I set the exit after relog to 30
rem
if errorlevel 30 goto loop
if errorlevel 1 goto end goto Loop: ←
```

IF YOU USE RELOGGING, YOU MUST ALSO ENABLE THE "RELOG" OPTION (REFER TO THE DESCRIPTION OF THAT OPTION ABOVE).



**AFTER UPLOAD EXIT <number>**

Definition: .....Specifies that Opus V1.70 is to exit at logoff with this DOS errorlevel if a user has uploaded any files. Could be useful for virus scanning.

Range: .....DOS errorlevel specification.

Required: .....No

Example: .....AFTER UPLOAD EXIT 50

**ALT <function key> <ax> <bx> <cx> <dx>**

Definition: .....Specifies the values to be loaded into the AX,BX,CX, and DX registers from the local keyboard when the specified ALT function key is pressed. This option requires knowledge of FOSSIL programming and is not intended for the average sysop.

Range: .....Function keys F1 through F10. Register values 0000-FFFF (values are hexadecimal).

Required: .....No

Example: .....ALT F1 8700 0000 0000 0000

**DEFINE <priv> CUME <number>**

Definition: .....Specifies the total number of minutes of online time that a user with this privilege may have during one 24 hour period spanning midnight to midnight. The time is cumulative from session to session and is reset to zero when the login date changes from the previous login date.

Note: this will not apply to a user with ASSTSYSOP or SYSOP privileges as their time is reset after each call.

Range: .....<priv> Opus privilege level name. <number> 10-32767 minutes.

Required: .....Yes, for all Opus privilege levels from TWIT to SYSOP

Examples: .....DEFINE DISGRACE CUME 45 DEFINE NORMAL CUME 80

**DEFINE <priv> FILE BAUD <number>**

Definition: .....Specifies the slowest bps (bits per second) rate that a user of the specified privilege level may use to transfer files.

Range: .....<priv> Opus privilege level name. <number> 300-38400 bps.

Required: .....Yes, for all Opus privilege levels from TWIT to SYSOP.

Examples: .....DEFINE NORMAL FILE BAUD 1200 DEFINE SYSOP FILE BAUD 300

**DEFINE <priv> FILE LIMIT <number>**

Definition: ..... Specifies the total amount of data that a user with the specified privilege may download during one 24 hour period spanning midnight to midnight.

Range: ..... <priv> Opus privilege level name.  
<number> 0-32767 Kbytes (where 1 Kbyte = 1024 bytes).

Required: ..... Yes, for all Opus privilege levels from TWIT to SYSOP.

Examples: ..... DEFINE DISGRACE FILE LIMIT 0  
DEFINE PRIVIL FILE LIMIT 500

**DEFINE <priv> LOGON BAUD <number>**

Definition: ..... Specifies the slowest bps (bits per second) rate that a user of the specified privilege level may use to logon.

Range: ..... <priv> Opus privilege level name. <number> 300-38400 bps.

Required: ..... Yes, for all Opus privilege levels from TWIT to SYSOP.

Examples: ..... DEFINE TWIT LOGON BAUD 19200 DEFINE DISGRACE LOGON BAUD 1200  
DEFINE PRIVIL LOGON BAUD 300

**DEFINE <priv> RATIO <ratio>**

Definition: ..... Specifies the file download/upload ratio permissible for this privilege level. Ratio is not enforced by Opus until the user has downloaded 100k bytes.

Range: ..... Specified as dd:uu, where:  
dd = 1-255 uu = 1-255

Required: ..... No. You may specify ratios only for those privilege levels which you wish to have enforced.

Examples: ..... DEFINE NORMAL RATIO 5:1 DEFINE PRIVIL RATIO 10:1

**DEFINE <priv> TIME <number>**

Definition: ..... Specifies the number of minutes of online time that a user with this privilege may have during one session.

Range: ..... <priv> Opus privilege level name. <number> 10-32767.

Required: ..... Yes, for all Opus privilege levels from TWIT to SYSOP.

Examples: ..... DEFINE DISGRACE TIME 30  
DEFINE NORMAL TIME 60

**EDIT DISABLE USERLIST**

Definition: ..... Specifies that the user will not have access to the user list when writing a message. The user list function is invoked if a user types '?' at the TO: prompt. This is a security option for sysops who may have a security reason for keeping the user list private.

Required: ..... No

## END SESSION SECTION

Definition: .....Specifies end of the session section of the control file.

Required: ..... Yes

## EUROPE

Definition: .....This specifies that Opus V1.70 will display birth dates and current dates as Day, Mon Year instead of Mon Day, Year.

Required: .....No

## EXPIRE DAYS <number>

Definition: .....Specifies the number of days prior to expiration by date that will constitute a warning period. During the warning period, the user will be shown the appropriate warning text file each time they logon. This specification only applies to users that have expire by date enabled in their user record in the Opus user file.

Range: ..... 1-32767

Required: .....No

Example: .....EXPIRE DAYS 30

## EXPIRE MINUTES <number>

Definition: .....Specifies the number of minutes prior to expiration by usage time that will constitute a warning period. During the warning period, the user will be shown the appropriate warning test file each time they logon. This specification only applies to users who have expiration by usage minutes enabled in their user record in the Opus user file.

Range: ..... 1-32767

Required: .....No

Example: .....EXPIRE MINUTES 180

## EXPIRE PRIV <priv>

Definition: .....Specifies the privilege level to which a user will be demoted, if their account expires by date or time and they have a "demote user" flag set in their user record.

Range: .....Opus privilege level name.

Required: .....No

Example: .....EXPIRE PRIV Disgrace

**EXTERNAL PROTOCOL <programname>**

Definition: .....Specifies the name of an external program to use for file download and upload.

Range: .....DOS file specification. In Opus V1.70, you may specify up to sixteen external file protocols.

Required: .....No

Example: .....EXTERNAL PROTOCOL c:\opus\kermit.exe

**FILE DATE AUTOMATIC**

Definition: .....Specifies that Opus will automatically display the DOS file date with the filename and description when files are listed by the user in the files areas. This also enables the Opus "new files" feature which displays an asterisk by any file that is newer than the user's last logon date.

Required: .....No

**FILE DATE <format>**

Formats: .....FILE DATE DD MMM YY  
FILE DATE DD MMM YYYY  
FILE DATE MM-DD-YY  
FILE DATE MM/DD/YY  
FILE DATE YYMMDD  
FILE DATE NONE

Definition: .....Specifies that the DOS file date will be inserted into the files database when a file is uploaded. The format of the date is specified by the fields.

Required: .....No

Examples: .....YYMMDD	inserts 910714
MM/DD/YY	inserts 07/14/91
MM-DD-YY	inserts 07-14-91
DD MMM YY	inserts 14 Jul 91
DD MMM YYYY	inserts 14 Jul 1991
NONE	Opus will not display a file date when user requests as area file listing

**FILE RAW <priv>**

Definition: .....Specifies what privilege Opus V1.70 requires for users to view/download files that are not listed in the files database.

Range: .....Opus privilege level name.

Required: .....No. If not specified, defaults to Normal.

Example: .....FILE RAW Extra

**FILE WHO <priv>**

Definition: ..... Specifies that Opus V1.70 should allow users of this privilege level or above to see the name of who uploaded a file when using the new "long" file listing menu option, doing a Locate or new file files listing.

Range: ..... Opus privilege level name.

Required: ..... No

Example: ..... FILE WHO Disgrace

**FORMAT DATE <specification>****FORMAT TIME <specification>**

Definition: ..... When Opus needs to display a time or date, you have control over its appearance. The method will be familiar to "C" programmers. It should be fairly easy for everybody else.

TO "C" PROGRAMMERS: This just \*looks\* similar to 'printf()' commands. It is not. Do not use '%' commands other than the ones listed.

In a nutshell, you will give Opus a "control string" to use in displaying the time and date. Each piece of a time or date is represented by a two character control command.

All control commands begin with the percent sign.

These control strings CAN contain almost any other text type information. Anything other than the control commands will be used verbatim.

ALL CONTROL COMMANDS MUST BE UPPERCASE CHARACTERS.

Here is a list of the available commands:

%A	display "am" or "pm" as appropriate
%B	display the month as a digit
%C	display the month as a 3-character abbreviation
%D	display the day
%E	display the hour (12-hour clock)
%H	display the hour (24-hour clock)
%M	display the minutes
%S	display the seconds
%Y	display the year (2-digits)
%%	display a percent sign

Required: ..... Yes

Examples: .....	Control string	Sample output	Notes
	%D-%C-%Y	09-Mar-87	garden variety
	%Y%B%D	870309	scientific
	%D/%B/%Y	09-03-87	European
	%D:%B:%Y	09:03:87	Japanese
	%H:%M:cst	14:30cst	24-hour clock
	%E:%M%A	02:30PM	12-hour clock
			no date wanted!



### **KILL PRIVATE <when>**

Definition: ..... Specifies the optional deletion of private messages after a user reads them.  
NEVER means that Opus will not delete private messages after receipt.  
ALWAYS means that Opus will always delete a private message after it is read.  
ASK means that Opus will prompt the user whether to delete a private message after it is read.

Required: ..... Yes, but only one of the options.

Examples: ..... KILL PRIVATE NEVER  
KILL PRIVATE ASK  
KILL PRIVATE ALWAYS

### **LASTREAD**

Definition: ..... Specifies that Opus V1.70 should use/update the file LASTREAD in message areas for the first user in the user list. Otherwise it is not done.

Required: ..... No, but may be needed for some external message editors used by sysops.

### **LISTSYSOP <privilege>**

Definition: ..... Specifies the privilege level equal to or above which a user must have if OPus V1.70 is to show the user Sysop.  
names when listing boards in the Matrix/Netmail message area (version 7 nodelist files required).

Required: ..... No

**LOCAL EDITOR <doscommand> %s**

Definition: .....Specifies the use of an external editor with Opus. If you use an external message editor, it will be available to anybody who has a privilege equal to SYSOP or HIDDEN. When replying to messages, Opus will quote the entire previous message using > NN: where NN are your initials.

Required: .....No

Example: .....To specify an external editor in keyboard mode, put this in your control file:

```
LOCAL EDITOR <doscommand> %s
```

The "<doscommand>" is the name of the program with any parameters. You cannot call a batch file here.

The "%s" is required. Opus will substitute a file name where it finds the "%s". For example, if your editor takes a command line switch "-f" for filename, you could put "-f%s" in the control file line:

```
LOCAL EDITOR C:\Brief\B -t -z -r -a -e -L43 %s
```

If you want to use your external message editor ON- LINE as well as in keyboard mode, put an exclamation point in front of the <doscommand> as in:

```
LOCAL EDITOR !D:\Myeditor %s
```

If you absolutely must use a batch file, you can do something like this:

```
Local Editor C:\COMMAND.COM editor.Bat %s
```

**TECHNICAL NOTE:**

The "%s" is not absolutely necessary. You could use MSGTMP.TMP instead. In any event, Opus will look for a file called "MSGTMP.TMP" in the default sub-directory when you exit your editor.

The program flow looks like this:

1. execute the program named as the external editor program.
2. convert MSGTMP.TMP to a message.

The point is that you do not actually have to go to an editor. You can execute any program. If MSGTMP.TMP exists when the program finishes, Opus will turn it into a message.

**LOGOFF MESSAGES**

Definition: .....Specifies that Opus V1.70 should ask the caller whether to leave a message to the sysop at logoff whether or not the sysop has configured a message area number zero.

Required: .....No. If message area 0 exists and has a message path, Opus will ask for logoff messages with or without this setting in the control file.

**LOGON ADDRESS**

Definition: .....Specifies whether Opus V1.70 will ask new callers for their street address when they logon. The result will be stored in the street address field of the user's record in the Opus user file.

Required: .....No

## LOGON ASCIIMENU

Definition: ..... Specifies that Opus V1.70 should include an option in the new user logon question to use ASCII menus. Everything else is the same as if the user had selected G)raphics. If you use this option, you need to include it in the CONFHELP.BBS file shown to new users at logon to help with the configuration of Opus to suit their hardware/software needs.

It is the same as choosing the G)raphics option, except that ASCII menus will be turned on and the help level will be set to EXPERT so that the normal Opus menus are not also shown.

Required: ..... No

## LOGON ASKALIAS

Definition: ..... Specifies that Opus V1.70 is to assume that the user gave the opposite when they logged on (ie, real name) and Opus should ask for their alias. Opus will ask "What is your ALIAS:". The result will be stored in the alias user field of the caller's user record in the Opus user file. Opus will check to make sure that the user is not trying to use the "real" name of any other user listed in your Opus user file. See the Opus logon sequence depicted in section 5 of this manual.

Required: ..... No

## LOGON ASKCITY

Definition: ..... Specifies whether Opus V1.70 will ask new callers for their city/state when they logon. The result will be stored in the city/state field of the user's record in the Opus user file.

Required: ..... No

## LOGON ASKPHONE

Definition: ..... Specifies whether you want Opus to ask a first time caller for their phone number which Opus will then store in the caller's user record in the Opus user file. See the Opus logon sequence depicted in section 5 of this manual.

Required: ..... No

## LOGON ASKREAL

Definition: ..... Specifies that Opus V1.70 is to assume that the user gave the opposite when they logged on (ie, an alias) and Opus should ask for their real name. Opus will ask "What is your REAL name:". The result will be stored in the alias user field of the user's record in the Opus user file.

Required: ..... No

## LOGON BIRTHDAY

Definition: ..... Specifies whether Opus V1.70 will ask new callers for their birth date when they logon. The result will be stored in the birth date field of the user's record in the Opus user file.

Required: ..... No

### LOGON CANALIAS

Definition: .....Specifies that Opus V1.70 will not ask new users at logon for an alias at logon, but will allow the aliases as a menu option on the History menu.

Required: .....No

### LOGON HANDHOLD

Definition: .....Specifies whether Opus V1.70 should drop a user, who has called fewer than 25 times AND makes more than 10 errors at menus, back to NOVICE menus, TTY graphics, turn more? on and screen clear off.

Required: .....No

### LOGON HIDE PASSWORD

Definition: .....Specifies whether Opus V1.70 will encode the user's password which is stored in user record in the Opus user file. This should be fairly useful for people who cannot guarantee how secure their user file is because Opus is running on a LAN or in a public room.

NOTE: If choosing this option with an existing user file, you must run HIDEPWD.EXE to encode the existing passwords.

Required: .....No

### LOGON KEYS <keys>

Definition: .....Specifies the default keys which Opus V1.70 should automatically assign to new callers.

Range: .....Opus key/lock specification.

Required: .....No

Example: .....LOGON KEYS DGIUO

### LOGON LEVEL <priv>

Definition: .....Specifies the default logon privilege level given to new callers, if you allow any new callers (see LOGON PREREGISTERED).

Range: .....Opus privilege level name.

Required: .....No

NOTE: .....cannot be used simultaneously with LOGON PREREGISTERED (see below).

Example: .....LOGON LEVEL Disgrace



**LOGON MYQUESTION <filename>**

Definition: .....Specifies whether Opus V1.70 will ask new callers the question which you have setup in the specified file. The result will be stored in the answer field of the user's record in the Opus user file.

NOTE: .....You should not use OECs to store this answer. Opus will do this for you automatically.

Range:.....Opus text file specification.

Required:.....No

Example: .....LOGON MYQUESTION c:\opus\logs\myquest

**LOGON NOFLAGS**

Definition: .....Specifies whether Opus V1.70 should allow a user, even a new user, to enter their name plus a string of configuration commands at logon. These commands will over-ride whatever (if anything) has been stored in the user's record in the Opus user file. If the user is a new user, Opus will bypass the "Mirror shades" questions entirely. Everything else will remain the same.

Doug might call your board and log on something like this:

```
Doug Boone /vo/he /l24 /w78 /my /fy /+y
```

Opus would take him around through all the new user logon rigmarole, and then parse all the '/' commands and set him up with Avatar graphics, help level of EXPERT, screen length 24, width 78, more? on, OpEd editor, and show him the time remaining on the menu prompts.

If he called again with another program that did not support Avatar graphics, he could call back with, "Doug Boone /va" and set his graphics to ANSI right from the start.

The full set of commands available are:

/V [O] [A] [P]	Video [Opus][ANSI][Plain]
/H [N] [R] [E] [H]	Help Level [NOVICE][REGULAR][EXPERT] or [HITECH]
/L [a number]	Length of terminal
/W [20-132]	Width of terminal
/F [Y] [N]	FullScreen Editor [Yes][No]
/K [Y] [N]	HotKeys [Yes][No]
/T [Y] [N]	Talker [Yes][No]
/+ [Y] [N]	+Time Remaining [Yes][No]
/M [Y] [N]	More [Yes][No]

The commands can all be strung together or have spaces in between, the '/' marks the beginning of the next command. If a user does NOT give either a 'Y' or 'N' for a flag, it will default to 'yes' and turn that command ON.

Opus will NOT check to see if the user would have access to the configuration menu or any of these particular commands. Why a Sysop would NOT allow users to set the width of their terminal or turn more on or off has always escaped Doug, so if you do not want to allow users to change their configuration this way, enable LOGON NOFLAGS.

Required:.....No



**LOGON NOPASSWORD <priv>**

Definition: ..... Specifies that Opus V1.70 will allow users of this privilege level ONLY to logon without asking for a password. Use this with your LOGON LEVEL <PRIV> to allow new users to log on without giving any password. However, the user's privilege will have to be equal to the LOGON Nopassword privilege level. Exactly. Any other privilege level will require a password. Users will be able to enter a password from the History menu so you can upgrade them later.

Range: ..... Opus privilege specification.

Required: ..... No

Example: ..... LOGON NOPASSWORD Disgrace

**LOGON PREREGISTERED**

Definition: ..... Specifies that no new callers will be allowed to logon to your system unless they are already listed in the Opus user file (ie, you must add any new callers to your Opus user file manually).

**Notes**

- (1) Cannot be used simultaneously with the LOGON LEVEL option (see above).
- (2) The first time a pre-registered user logs on, Opus will not ask for the user password if one exists.

Required: ..... No

**LOGON SECTION <section>**

Definition: ..... Specifies the default section which Opus V1.70 should automatically assign to new callers.

Range: ..... Opus section specification.

Required: ..... No

Example: ..... LOGON SECTION A

**LOGON TIMELIMIT <number>**

Definition: ..... Specifies the time allowed for logon (ie, reaching the main menu) by any caller. The purpose of limiting the logon time is to hang up on calls that do not complete logon properly. An example would be a user calling with a buggy automated terminal script that gets hung in a wait state. The limited logon time will cause Opus to hang up, thus freeing up the system for the next caller in a reasonable time.

Range: ..... 10 minutes or greater

Required: ..... No, defaults to 10 minutes.

### LOGON XDAYS <number>

Definition: .....Specifies the default number of days that Opus V1.70 should credit to new callers. After the specified number of days, their access to the system will expire (see also EXPIRE PRIV below).

Range: ..... 1-32767

Required: .....No

Example: .....LOGON XDAYS 7

### LOGON XMINs <number>

Definition: .....Specifies the default number of minutes that Opus V1.70 should credit to new callers. After the specified number of minutes, their access to the system will expire (see also EXPIRE PRIV below).

Range: ..... 1-2147483647

Required: .....No

Example: .....LOGON XMINs 300

### MAKESECTION <section>

Definition: .....Specifies the section that Opus V1.70 will temporarily dump users into while in the system the edits the User Defined Section (UDS). Otherwise, users would not be able to add any new areas to their UDS.

Range: .....Opus section specification.

Required: ..... Yes, if you allow users to create their own UDS and do not want users to have access to ALL areas.

Example: .....MAKESECTION A

### MIN LOGON BAUD <number>

Definition: .....Specifies the slowest speed caller that you wish to allow to logon to your BBS.

Range: ..... 1200, 2400, 4800, 7200, 9600, 19200, 38400

Required: .....No. This option is not necessarily recommended by anyone associated with Opus development.

Example: .....MIN LOGON BAUD 1200

**MIN NONTTY BAUD <number>**

Definition: .....Specifies the slowest speed at which you will allow a caller to use graphics such as ANSI or AVATAR.

Range: .....1200, 2400, 4800, 7200, 9600, 19200, 38400

Required: .....No. May be useful to prevent low speed callers from inadvertently selecting a graphics option that would be too excruciatingly slow to be useable.

Example: .....MIN NONTTY BAUD 2400

**REFUND <number>**

Definition: .....Specifies the percentage time rebate to give users when they upload for that logon session only and provided it would not overrun a forced event.

Range: .....0-32767

Required: .....No (default is 100 percent)

Example: .....REFUND 200 would give a 200 percent refund, that is, for every minute spent uploading, a user would receive two minutes additional online time.

**RELOG <number>**

Definition: .....Specifies how users can re-enter Opus after they have exited and your batch file cycles up to re-start Opus with the '-o' command line Option. If successful, user will return to the exact point where they left off.

Range: .....0-3

- 0 Do not allow relogging on. Hang up.
- 1 Ask user for name and password and check it against the LASTUS##.DAT file and USER.DAT file.
- 2 Ask user for Password only and check it against LASTUS##.DAT file and USER.DAT file.
- 3 Do not ask any questions. Just return user to where they were.

Required: .....Yes if you plan to use the '-o' option to restart Opus after exiting while a user is online.

IF YOU USE RELOGGING YOU MUST ALSO ENABLE THE "AFTER RELOG" EXIT OPTION (REFER TO THE DESCRIPTION OF THAT OPTION BELOW).

**SECURE <number>**

Definition: .....Specifies the action Opus will take with the LASTUS##.DAT file when returning from an external program via the RUN/SRUN or DOS/SDOS methods. The options are:

- 0 Never re-read LASTUS##.DAT, ignore it.
- 1 Just copy the information from OPUS\_id on, no OPUS information is copied
- 2 Check the Name, Privs, Keys, and OPUS\_id before copying the complete user record.
- 3 Just check the OPUS\_id and live dangerously.

Range: .....0-3

Required: .....No

Example: .....SECURE 2

**SHOWTIME**

Definition: .....Specifies that Opus V1.70 should allow users to show the time remaining at their menu prompts. If this is not enabled, Opus will not even ask them about it.

Required: .....No

**USES BADNAME <filename>**

Definition: .....Specifies the name of the file that Opus V1.70 will show to a caller whose name has been rejected by the Opus NAMEFILTER (see above).

Range: .....Opus text file specification.

Required: .....No

Example: .....USES BADNAME c:\opus\misc\badname

**USES BADPASSWORD <filename>**

Definition: .....Specifies the name of the file Opus V1.70 will show to users when they have failed to correctly enter their password 5 times. Clever use of OECs should enable you to collect sufficient information to re- verify the user or tell them how to contact you to establish their identity.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES BADPASSWORD c:\opus\misc\bad\_pwd

**USES BARRICADE <filename>**

Definition: .....Specifies the name of the file shown to a user when they attempt to access a barricaded area. This file should explain that a password is required to enter the area, that if they have stumbled into the area by mistake, they should press ENTER to return to the system and that Opus will hang up on them if they do not enter the correct password.

**Format of file:**

```
Warning! Warning! ^G^G^G
This is a password protected area.
If you have stumbled in here by mistake,
press ENTER or RETURN to return to the system.
Any other action will be recorded as an attempt
to enter this area and you will be disconnected.
```

Range: .....Opus text file specification.

Required: .....No

Example: .....USES BARRICADE c:\opus\misc\baricade

**USES BYEBYE <filename>**

Definition: .....Specifies the name of a file to show to users after they have chosen to logoff. This file is shown just before the Opus built-in prompt instructing users to hang up.

Range:.....Opus text file specification.

Required:.....No

Example: .....USES BYEBYE c:\opus\misc\byebye

**USES CONFIG <filename>**

Definition: .....Specifies the name of the file that Opus V1.70 will show to new callers to help them through the traumatic new user logon configuration process. An expanded "mirror shades" question, users can select four choices. This file will be shown before they make the choice and can be recalled if they select help. See the Opus logon sequence depicted in section 5 of this manual.

Range:.....Opus text file specification. Caution should be used: at this point in the new user configuration process you do not know anything about the new user (eg, graphics capability etc).

Required:.....No

Example: .....USES CONFIG c:\opus\misc\shades

**USES CONFMENU <filename>**

Definition: .....File displayed to user as fancy, slow ASCII menu at the CONFIG menu if the user has ASCII menus turned on. Opus menu will follow.

Range:.....Opus text file specification.

Required:.....No

**USES CONTENTSHELP <filename>**

Definition: .....Specifies text file to show to the user when the File Contents command is invoked.

Range:.....Opus text file specification.

Required:.....No

Example: .....USES CONTENTSHELP c:\opus\hlp\contents

**USES CUSTOM<n>MENU <filename>**

Definition: .....File displayed to user as fancy, slow ASCII menu at the specified custom menu if the user has graphic menus turned on. The normal custom menu will follow.

Range:.....<n> 1-6 <filename> Opus text file specification.

Required:.....No

Example: .....USES CUSTOM4MENU c:\opus\menu\custom4



### **USES DAYLIMIT <filename>**

Definition: .....Specifies the name of the file to show to a user when they have exceeded their daily total time limit.

Range: .....Opus text file specification.

Required: ..... Yes

Example: .....USES DAYLIMIT c:\opus\misc\daylimit

### **USES DOWNLOAD <filename>**

Definition: .....Specifies the file to which Opus V1.70 will log all downloads.

Required: .....No

### **USES FILEAREAS <filename>**

Definition: .....Specifies text file to show to the user when the area list is requested in the file section in response to the Area Change command prompt. This file should present all available file areas to the user. It allows customized area listings in place of the default automatic area listing presented by Opus.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES FILEAREAS c:\opus\misc\filearea

### **USES HISTORYMENU <filename>**

Definition: .....Specifies the name of the file which Opus V1.70 will show to a caller before the History menu if the caller has tedious ASCII menus turned on.

Range: .....Opus text file specification.

Required: ..... Yes

Example: .....USES HISTORYMENU c:\opus\gmenus\history

### **USES INQUIREHELP <filename>**

Definition: .....Specifies the text file to show when the user requests the inquire function in the message section.

Range: .....Opus text file specification.

Required: ..... Yes

Example: .....USES INQUIREHELP c:\opus\hlp\inquire

### USES INTERLINE <filename>

Definition: .....Specifies the hel file for the Opus V1.70 interline message module (the \_SEND\_MESSAGE menu command).

Range: .....Opus text file specification.

Required: .....No

Example: .....USES INTERLINE c:\opus\misc\userchat

### USES LEAVING <filename>

Definition: .....Specifies the name of a text file to show to the user when an external program is called from Opus via the RUN or DOS menu methods.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES LEAVING c:\opus\misc\leaving

### USES LOCATEHELP <filename>

Definition: .....Specifies text file to show to the user when the File Locate command is invoked.

Range: .....Opus text file specification

Required: .....No

Example: .....USES LOCATEHELP c:\opus\hlp\locate

### USES LOGO <filename>

Definition: .....Specifies the name of the file shown to callers when they connect to your BBS. This should be a pure seven bit printable ASCII text file. Opus will filter the file to eliminate rogue characters, which could cause problems with mailer connections. See the Opus logon sequence depicted in section 5 of this manual.

Range: .....LOGO.BBS should not contain ANY graphics or control characters. Opus will stop showing the file when it hits any and log them thus: LOGO: Invalid Char (x/'x').

NOTE: .....The LOGO.BBS file should be kept very short or it may hinder, or even prevent, the negotiation of a successful mail session with another mailer.

Required: .....No

Example: .....USES LOGO c:\opus\misc\logo

### **USES LOREHELP <filename>**

Definition: .....Specifies the text file to be shown to users who have posted fewer than 2 messages when using the line oriented editor (LORE). The following is a suggested help file:

```
Opus "wraps" lines automatically - there is no need to press ENTER
at the right margin.
Press ENTER twice to quit the editor.
<SPACE><RETURN> creates a blank line between paragraphs.
```

Range: .....Opus text file specification.

Required: .....No

Example: .....USES LOREHELP c:\opus\hlp\editor

### **USES MACRO <filename>**

Definition: .....Specifies the name of the help file Opus V1.70 will show to users when they set about defining their user macros.

Range: .....Opus text file specification.

Required: .....Yes

Example: .....USES MACRO c:\opus\hlp\macro

### **USES MAINMENU <filename>**

Definition: .....File displayed to user as fancy, slow ASCII menu at the MAIN menu if the user has ASCII menus turned on. Opus menu will follow.

Range: .....Opus text file specification.

Required: .....No

### **USES MAKE <filename>**

Definition: .....Specifies the name of the help file Opus V1.70 will show to users when they set about defining their own UDS (user defined section).

Range: .....Opus text file specification.

Required: .....Yes

Example: .....USES MAKE c:\opus\hlp\makesecs

**USES MAYBENEW <filename>**

Definition: .....Specifies the name of the file which Opus V1.70 will show to callers when they try to log on between the "Can't find" and "Register as new user [y,N,?]" logon prompts. The idea is that you can tell users about your policy on handles (knobs?), or that you do not allow new users or how to log on as "guest" or whatever. Remember that you do not know anything about this person and so you do not know if they have graphics capability or the size of their screen. See the Opus logon sequence depicted in section 5 of this manual.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES MAYBENEW c:\opus\misc\newbie

**USES MSGAREAS <filename>**

Definition: .....Specifies the text file to show to the user when an area listing is requested in the message section. This file should present all available message areas to the user. It allows customized area listings in place of the default automatic message area listing presented by Opus and is significantly faster.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES MSGAREAS c:\opus\misc\msgarea

**USES NAMEFILTER <filename>**

Definition: ..... Specifies the name of the file that Opus V1.70 will use for its "Trashcan" filter file (see also USES BADNAME below). The file contains a sequence of lines that are interpreted in one of two ways, depending on the first character:

? the material that follows on this line is not to appear in any form in a user's name

EXAMPLE: ?shit

! the material that follows on this line is not to appear AS IS in a user's name.

This is a case sensitive filter.

EXAMPLE: !Ass

If a line in the file begins with some character other than "?" or "!" then the entire line is ignored. There is no error message.

Remember that Opus converts all names to an uppercase character followed by lower-case characters. In the first example, the character sequence "shit" would be disallowed in any form in a user's name. In the second example, "Ass" would be thrown out ONLY if it appears as the first Three characters of a user's first or last name. The reason for the second (case sensitive) version is this: the characters "ass" appear in several legitimate names.

One other special character is available: the pound sign. Opus will convert a pound sign (#) to a SPACE character. This lets you further refine your list of banned words.

EXAMPLE: ?All#

After all, you do not want to keep everybody named "Allison" off your system just because you are trying to keep folks from logging on as "All".

If USES NAMEFILTER has been specified, Opus will go through the file on line-by-line after a new caller enters his/her name. If Opus finds a problem, it will do one of two things:

- (1) pretend you run a PRIVATE system that requires pre-registration; or
- (2) show the file specified by the USES BADNAME option (see below)

Problems include a match with any item in the file ... or some DOS I/O error in reading the file.

Required: ..... No

Example: ..... USES NAMEFILTER c:\opus\misc\username.txt

**USES NEWUSER1 <filename>**

Definition: ..... Specifies the name of the first file shown to a first time caller after they have answered the Opus first name and last name prompts and before Opus prompts them to choose a password. The distributed NEWUSER1.BBS contains help on how to choose a password. This file is shown to first time callers only. See the Opus logon sequence depicted in section 5 of this manual.

Range: ..... Opus text file specification.

Required: ..... No

Example: ..... USES NEWUSER1 c:\opus\misc\newuser1



**USES NEWUSER2 <filename>**

Definition: .....Specifies the name of the file shown to first time callers after they choose a password and before the WELCOME file. This file is shown to first time callers only. See the Opus logon sequence depicted in section 5 of this manual.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES NEWUSER2 c:\opus\misc\newuser2

**USES OPED <filename>**

Definition: .....Specifies the text file to show to a user when the user requests help from the Full Screen Message Editor (OPED). Automatically shown when entering OpEd if the user has entered fewer than 2 messages.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES OPED c:\opus\hlp\oped

**USES QUOTE <filename>**

Definition: .....Specifies the name of the Quotes file. This file contains ASCII text delimited by single empty lines. The material shown between empty lines is treated as one "quote" and is displayed by the Opus embedded command for quote. Opus keeps a position pointer for the quotes file to show quotes in a "round robin" order.

Range: .....Opus text file specification. Note that Opus will only use the .BBS extension for this file.

Required: .....No

Example: .....USES QUOTE c:\opus\misc\quotes

**USES REPLACEHELP <filename>**

Definition: .....Specifies the text file to show when the user requests the edit line function from the Line Oriented Message Editor (LORE).

Range: .....Opus text file specification.

Required: .....Yes

Example: .....USES REPLACEHELP c:\opus\hlp\rep\_edit

### **USES RETURNING <filename>**

Definition: .....Specifies the name of a text file to show to the user when returning to Opus from an external program invoked by the RUN or DOS methods.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES RETURNING c:\opus\misc\return

### **USES ROOKIE <filename>**

Definition: .....Specifies the name of the file shown to a new caller on the second and third logon call. It will be shown just prior to the welcome screen. See the Opus logon sequence depicted in section 5 of this manual.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES ROOKIE c:\opus\misc\rookie

### **USES SECTMENU <filename>**

Definition: .....File displayed to user as fancy, slow ASCII menu at the SECTION menu if the user has ASCII menus turned on. Opus menu will follow.

Range: .....Opus text file specification.

Required: .....No

### **USES SYSOPMENU <filename>**

Definition: .....File displayed to user as fancy, slow ASCII menu at the SYSOP menu if the user has ASCII menus turned on. Opus menu will follow.

Range: .....Opus text file specification.

Required: .....No

### **USES TIMEWARN <filename>**

Definition: .....Specifies the name of the warning file to show to a user when they have only 2 minutes left in their current online session.

Range: .....Opus text file specification.

Required: .....Yes

Example: .....USES TIMEWARN c:\opus\misc\timewarn

**USES TOOSLOW <filename>**

Definition: .....Specifies the name of the file to show to a user when they have logged on using a baud rate below the minimum specified in LOGON BAUD (see above).

Range: .....Opus text file specification.

Required: ..... Yes

Example: .....USES TOOSLOW c:\opus\misc\tooslow

**USES UPLOAD <filename>**

Definition: .....Specifies the file to which Opus V1.70 will log all uploads.

Required: .....No

**USES WELCOME <filename>**

Definition: .....Specifies the name of the WELCOME file shown to all users unless a Special OECC file has been defined in the caller's user record. See the Opus logon sequence depicted in section 5 of this manual.

Range: .....Opus text file specification.

Required: ..... Yes

Example: .....USES WELCOME c:\opus\misc\welcome

**USES XDGONE <filename>**

Definition: .....Specifies the name of the file to show to a user when they have reached the expiration date set in their user record in the Opus user file.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES XDGONE c:\opus\misc\xdgone

**USES XDWARN <filename>**

Definition: .....Specifies the name of the warning file to be shown to a user when they are within the specified number of days (see EXPIRE DAYS below) of the expiration date set in their user record in the Opus user file.

Range: .....Opus text file specification.

Required: .....No

Example: .....USES XDWARN c:\opus\misc\xdwarn

### **USES XFERBAUD <filename>**

Definition: ..... Specifies text file to show to the user if their baud rate is not fast enough to do a file transfer (see also FILE BAUD below).

Range: ..... Opus text file specification.

Required: ..... No

Example: ..... USES XFERBAUD c:\opus\misc\xferbaud

### **USES XTGONE <filename>**

Definition: ..... Specifies the name of the file to show to a user when they have exhausted the number of minutes allotted to them in their user record in the Opus user file.

Range: ..... Opus text file specification.

Required: ..... No

Example: ..... USES XTGONE c:\opus\misc\xtgone

### **USES XTWARN <filename>**

Definition: ..... Specifies the name of the warning file to show to a user when they are within the specified number of minutes (see EXPIRE MINUTES below) of using up the time allocated to them in their user record in the Opus user file.

Range: ..... Opus text file specification.

Required: ..... No

Example: ..... USES XTWARN c:\opus\misc\xtwarn

### **USES YELL <filename>**

Definition: ..... Specifies the name of the file that Opus V1.70 will show to users if the CHAT fossil program is not loaded or Yell is turned off or it is after hours. You can use this file to run any other external Chat program via an OEC.

Range: ..... Opus text file specification.

Required: ..... No

Example: ..... USES YELL c:\opus\misc\yell

### 2.8.6 AREA <number> <string>

Definition: .....Begins the definition of an area in Opus V1.70 by specifying the area number and, optionally, the area name which users can enter at the change area prompt instead of the area number. If specified, the area name will also be shown to users between the area number and file area TITLE or message area TITLE.

Range: .....<number> 0-32767. <string> Up to 16 printable ASCII characters.

Required: .....Area 0 and Area 1 must be defined. An AREA statement must be used at the beginning of each area defined in the Opus control file.

Example: .....AREA 1 GEN

With the above example, assuming a message area, the following would appear in the area:

```
MESSAGE area    1  .. GEN                General Messages
```

A user could then type in either 1 or GEN at the change area prompt and be transported to this area.

#### ACCESS DOWN <priv>

OBSOLETE. (In Opus V1.70, see DOWNLOAD PRIV instead.)

#### ACCESS EDIT <priv>

OBSOLETE. (In Opus V1.70, see EDIT PRIV instead.)

#### ACCESS FILE <priv>

OBSOLETE. (In Opus V1.70, see ACCESS PRIV instead.)

#### ACCESS FILE\_EXTERN <priv>

OBSOLETE. (In Opus V1.70, a particular privilege level may be specified for each "external" menu option. See further section 3 of this manual.)

#### ACCESS LOCK <locks>

Definition: .....Specifies the lock settings on user access to this message or file area. A user must have the corresponding keys turned on in their Opus user file record for access to this message or file area.

Range: .....Opus key/lock specification.

Required: .....Optional for a defined message area.

Example: .....ACCESS LOCK AD05

The above example would turn on locks A, D, 0, and 5.

#### ACCESS MESSAGE <priv>

OBSOLETE. (In Opus V1.70, see instead ACCESS PRIV.)

#### ACCESS MSG\_EXTERN <priv>

OBSOLETE. (In Opus V1.70, a particular privilege level may be specified for each "external" menu option. See further section 3 of this manual.)



**ACCESS PRIV <priv>**

Definition: .....Specifies minimum user privilege to access this message or file area.

Range: .....Opus privilege level name.

Required: .....Yes, if message or file area is defined.

Example: .....ACCESS PRIV Disgrace

**ACCESS UP <priv>**

OBSOLETE. (In Opus V1.70, see UPLOAD PRIV instead.)

**ADDRESS <zone:net/node.point>**

Definition: .....Specifies the FULL zone:net/node.point address to use in this echomail message area for the Origin line, MSGID, and SEEN-BY line (net/node only used).

Range: .....zone:net/node.point specified as 1:1/1.0 through 32767:32767/32767.32767

Required: .....No

**ALIAS**

Definition: .....Specifies that users may create messages from their alias name in this area. Opus V1.70 will automatically use the user's alias name when they create a message.

Required: .....Yes, if you want users to be able create messages from their alias name in a message area.

**ANONYMOUS OKAY**

Definition: .....Specifies that users may create messages anonymously in this area. Opus will prompt the user to enter an alias name or use their own name when they create a message.

Required: .....Yes, if you want users to be able create messages anonymously in a message area.

**APPLICATION <string>**

Definition: .....Specifies information to be stored in the Opus V1.70 message (SYSMSG.DAT) or files (SYSFILE.DAT) area definitions in the message or files database files. If the line occurs in a message area definition, then it will be stored in SYSMSG.DAT; if the line occurs in a file area definition, then it will be stored in SYSFILE.DAT. The information stored is for use by external programs or "applications" which would have to read this information out of the relevant SYSMSG.DAT or SYSFILE.DAT file.

Range: .....Opus text specification.

Required: .....No

Example: .....APPLICATION RENUM -2 -N 1 100 -R

If specified in a message area definition in the Opus control file, the above example would cause NACL to store "RENUM -2 -N 1 100 -R" in the message area's definition.

**BARRICADE <filename>**

Definition: ..... Specifies the file that Opus V1.70 will use to determine passwords and access levels for this message or file area. The format of the file is:

```
PASSWORD  PRIVILEGE
OPEN      NORMAL
SEASAME   ASSTSYSOP
K9W3F1    SYSOP
```

You can also use names in your barricade files:

```
Doug Boone  TWIT
Joe Smith   NORMAL
```

Neither Doug Boone nor Joe Smith would have to give a password for the barricaded area. Instead, their privileges would be automatically set by Opus. Doug Boone would not get access to the area, even if someone else told him the barricade password for this area.

NOTE: ..... see also the USES BARRICADE option above.

Range: ..... DOS filename specification.

Required: ..... No

**CHARSET <identifier>**

Definition: ..... Specifies the ISO character set Opus V1.70 is to use when displaying messages in this message area.

Range: ..... The currently available character sets are:

- PC-8    The "standard" IBM character set, including high-bit graphics. Using this can be a good way to make enemies, but it is probably necessary in Asia to work with their Big-5 system.
- ISO-4    United Kingdom.
- ISO-6    ASCII
- ISO-11   Swedish
- ISO-15   Italian
- ISO-17   Spanish
- ISO-21   German
- ISO-60   Norwegian
- ISO-69   French
- ECMA-94   LATIN-1 (Amiga)

WARNING:        Except for the PC-8 character set any 8-bit characters will be translated to 7-bit characters before they are stored in the \*.MSG file. PC-8 goes straight through, including high-bit characters. You might use it on your local areas, but I guarantee we will hear about it if your users start using it to send around fancy advertising in their messages.

NOTE:    You must also define the character translation table (see the CHARSET <filename> option in the Language section above)

Example: ..... CHARSET ISO-60

### **DOMAIN <string>**

Definition: .....Specifies the InterNet domain that Opus V1.70 should use when inserting MSGIDs in messages in this echomail message area.

Range: .....Any valid InterNet domain.

Required: .....Not unless ECHO MSGID is enabled (see above).

Example: .....DOMAIN fidonet.org

### **DOWNLOAD <path\>**

Definition: .....Specifies the DOS path where downloadable files for this file area are located.

Range: .....DOS path specification.

Required: .....Yes, if file area defined.

Example: .....DOWNLOAD c:\opus\files\games\

### **DOWNLOAD LOCK <locks>**

Definition: .....Specifies the lock settings on user access to the download command in this file area. A user must have the corresponding keys turned on in their Opus user record for access to the command.

Range: .....Opus key/lock specification.

Required: .....Optional for a defined file area.

Example: .....DOWNLOAD LOCK BD04  
The above example would turn on locks B, D, 0, and 4.

### **DOWNLOAD PRIV <priv>**

Definition: .....Specifies the minimum user privilege required before Opus V1.70 will allow a user to download from this file area.

Range: .....Opus privilege level name.

Required: .....Optional for a defined file area.

Example: .....DOWNLOAD PRIV Worthy

### **DUPES**

Definition: .....Specifies that Opus V1.70 should check for duplicate files when uploads start in this file area. Opus will allow users who have a privilege level of Asstsysop or above to upload duplicate files, in which case Opus will add a counter to the filename extension. For example, if the sysop uploaded BOMB.ZIP and BOMB.ZIP already existed in that file area, Opus would rename the upload to BOMB.ZI1 and so on.

Required: .....No

**ECHO <path\>**

OBSOLETE. (In Opus V1.70, see instead:      PATH, LOCAL,      ECHOMAIL and MATRIX.)

**ECHOMAIL <tag>**

Definition: .....Specifies that messages in this area are ECHOMAIL.

There are three kinds of message areas:

- LOCAL means all messages are local to the BBS and are not sent to other systems.
- ECHOMAIL means messages are sent to other systems in shared echomail conferences with the same echomail tag (eg, the tag name for the Opus echomail support conference is MEADOW).
- MATRIX means the area where "one-to-one" messages which are sent to/from other systems are stored. You can have only one MATRIX area. This MATRIX area is also called the NetMail area in some circles.

You cannot use more than one of these statements in a single message area.

Range: .....<tag> Up to 32 printable ASCII characters; usually      upper case, but not checked by  
Opus (only      required for ECHOMAIL areas).

Required: ..... Yes, one type for each message area that is defined.

Examples: .....LOCAL  
ECHOMAIL MEADOW  
ECHOMAIL PHOTO  
MATRIX

**EDIT LOCK <locks>**

Definition: .....Specifies the lock settings that Opus V1.70 enforces on user access to the enter message and reply commands in this message area. A user must have the corresponding keys turned on in their Opus user file record for access to this message area.

Range: .....Opus key/lock specification.

Required: .....Optional for a defined message area.

Example: .....LOCK EDIT BD04  
The above example would turn on locks B, D, 0, and 4.

**EDIT PRIV <priv>**

Definition: .....Specifies the minimum user privilege Opus V1.70 requires before a user is allowed to enter or reply to a message in this message area.

Range: .....Opus privilege level name.

Required: .....Optional for a defined message area.

Example: .....EDIT PRIV Normal



## END AREA

Definition: ..... Specifies the end of an area definition.

Required: ..... Yes, at the end of each area defined in the control file.

## EXTERN MAILCHECK

Definition: ..... Specifies that Opus V1.70 is to store flags for an external mailchecking program in the message area definitions database file (SYSMSG.DAT). The information stored would have to be read out of the SYSMSG.DAT file by the external mailchecking program.

Required: ..... No

## EXTERN RENUM

Definition: ..... Specifies that Opus V1.70 is to store flags for an external message renumbering program in the message area definitions database file (SYSMSG.DAT). The information stored would have to be read out of the SYSMSG.DAT file by the external renumbering program.

Required: ..... No

## FILEINFO <description>

OBSOLETE. (In Opus V1.70, see TITLE instead.)

## FREE

Definition: ..... Tells Opus V1.70 that files downloaded from this file area are not to affect a caller's up:download ratio, daily limits or total downloads.

Required: ..... No

## HELP <filename>

Definition: ..... Specifies the help file to be shown by Opus V1.70 to users in the relevant message or file area. This will override the menu options F\_HELP and MSG\_HELP (see further section 3 of this manual).

Range: ..... DOS filename specification.

Required: ..... No

Examples: ..... HELP c:\opus\hlp\file31  
HELP c:\opus\hlp\msg31

## INBOUND ONLY

Definition: ..... Specifies that mail is imported into this echomail message area, but not scanned out to other nodes.

Required: ..... Yes, if this is an ECHOMAIL message area and you do not wish to send messages out to other nodes. You must specify either INBOUND ONLY or SCAN <nodes> for an ECHOMAIL message area.



**KUPLOAD <path\>**

Definition: ..... Specifies the DOS path where files uploaded for this file area will be stored by Opus V1.70.

- UPLOAD The "normal" default upload path.
- PUPLOAD The upload path for users with a privilege level greater than that specified in the PUPLOAD PRIV option.
- KUPLOAD The upload path for users who have the keys specified by the KUPLOAD LOCK option.

Range: ..... DOS path specification.

Required: ..... UPLOAD Yes, if a file area is defined. The path can be identical to the DOWNLOAD path or any other legal DOS path on the system.  
 PUPLOAD No  
 KUPLOAD No

Examples: ..... UPLOAD c:\opus\files\games\  
 PUPLOAD c:\opus\files\private\  
 KUPLOAD c:\opus\files\locked\

**KUPLOAD AREA <number>**

Definition: ..... Specifies the file area number of the file area that uploads will be considered to be a part of in the Opus V1.70 files database.

Range: ..... 0-32766, the NUMBER SHOULD MATCH THE FILE AREA NUMBER OF THE FILE AREA WHERE UPLOADS, PUPLOADS or KUPLOADS ARE STORED or I confidently predict that you will be in for some interesting times with your files database.

Required: ..... YES, ESSENTIAL for each upload, private upload or keyed upload path defined!

Examples: ..... UPLOAD AREA 31 PUPLOAD AREA 41 KUPLOAD AREA 51

**KUPLOAD LOCK <locks>**

Definition: ..... Specifies the lock settings that Opus V1.70 requires for user uploads in this file area to go to the KUPLOAD path. A user must have the corresponding keys turned on in their Opus user record before uploads will go to the defined KUPLOAD path.

Range: ..... Opus key/lock specification.

Required: ..... No. If specified, the KUPLOAD <path> control file option must also be set for this file area.

Example: ..... KUPLOAD LOCK DGIOU

**LIST <filename>**

OBSOLETE. (In Opus V1.70, the files database allows you to specify the full path to a file if it is not in the default directory for that file area. People with CD-ROMS and LAN servers will probably find this very useful).

## LOCAL

Definition: .....Specifies that messages in this area are LOCAL.

There are three kinds of message areas:

- LOCAL means all messages are local to the BBS and are not sent to other systems.
- ECHOMAIL means messages are sent to other systems in shared echomail conferences with the same echomail tag (eg, the tag name for the Opus echomail support conference is MEADOW).
- MATRIX means the area where "one-to-one" messages which are sent to/from other systems are stored. You can have only one MATRIX area. This MATRIX area is also called the NetMail area in some circles.

You cannot use more than one of these statements in a single message area.

Range:.....<tag> Up to 32 printable ASCII characters; usually upper case, but not checked by Opus (only required for ECHOMAIL areas).

Required:.....Yes, one type for each message area that is defined.

Examples:.....LOCAL

### LOCAL <path\>

OBSOLETE. (In Opus V1.70, see instead: PATH, LOCAL, ECHOMAIL and MATRIX.)

### LOCK DOWN <locks>

OBSOLETE. (In Opus V1.70, see DOWNLOAD LOCK instead.)

### LOCK EDIT <locks>

OBSOLETE. (In Opus V1.70, see EDIT LOCK instead.)

### LOCK FILE <locks>

OBSOLETE. (In Opus V1.70, see ACCESS LOCK instead.)

### LOCK FILE\_EXTERN <locks>

OBSOLETE. (In Opus V1.70, a particular sequence of locks may be specified for each "external" menu option. See further section 3 of this manual.)

### LOCK MESSAGE <locks>

OBSOLETE. (In Opus V1.70, see instead ACCESS LOCK.)

### LOCK MSGEXTERN <locks>

OBSOLETE. (In Opus V1.70, a particular sequence of locks may be specified for each "external" menu option. See further section 3 of this manual.)

### LOCK UP <locks>

OBSOLETE. (In Opus V1.70, see UPLOAD LOCK instead.)

**MATRIX**

Definition: .....Specifies that messages in this area are MATRIX.

There are three kinds of message areas:

- LOCAL means all messages are local to the BBS and are not sent to other systems.
- ECHOMAIL means messages are sent to other systems in shared echomail conferences with the same echomail tag (eg, the tag name for the Opus echomail support conference is MEADOW).
- MATRIX means the area where "one-to-one" messages which are sent to/from other systems are stored. You can have only one MATRIX area. This MATRIX area is also called the NetMail area in some circles.

You cannot use more than one of these statements in a single message area.

Required: ..... Yes, one type for each message area that is defined.

Examples: .....MATRIX

**MATRIX <path>**

OBSOLETE. (In Opus V1.70, see instead:      PATH, LOCAL,      ECHOMAIL and MATRIX.)

**MAXLINES <number>**

Definition: .....The 60 line limit on messages has also been lifted. Instead Opus V1.70 allows you to specify a maximum number of lines for messages in each message area. Opus will give users the number of lines specified, or as many as it can in the amount of free memory that it has available.

Range: .....The range of lengths that NACL will allow is 10 up to 250 lines.

Required: .....No, defaults to 60 lines.

Example: .....MAXLINES 100

**MENU <number>**

Definition: .....Specifies which menu number to use. It allows you to switch menus by area. Do not confuse this option with language switches.

Range: .....0-254. Use number 0, if you want to use the default menu. (See example).

Required: .....No

Example: .....Suppose you have six menus defined as:  
ENGLISH.MNU, ENGLISH.001, ENGLISH.002 FRENCH.MNU, FRENCH.001,  
FRENCH.002

Menu 0 means use ENGLISH.MNU or FRENCH.MNU. Menu 1 means use ENGLISH.001 or FRENCH.002. Menu 2 means use ENGLISH.002 or FRENCH.002.

Of course, the user's language choice would specify whether to use the ENGLISH or FRENCH menu groups.

**MSGINFO <description>**

OBSOLETE. (Replaced by TITLE in Opus V1.70.)

**MSGNAME <areaname>**

OBSOLETE. (In Opus V1.70, see instead ECHOMAIL.)

**NAME <string>**

Definition: .....A message or file area can have multiple names which users can enter at the change area prompt instead of the area number. These area names will not be shown to users between the area number and file area TITLE or message area TITLE.

Range: .....Up to 39 printable ASCII characters.

Required: .....Area 0 and Area 1 must be defined. An AREA statement must be used at the beginning of each area defined in the Opus control file.

Example: .....AREA 1 GEN

With the above example, assuming a message area, the following would appear in the area:

```
MESSAGE area 1 .. GEN                General Messages
```

A user could then type in either 1 or GEN at the change area prompt and be transported to this area.

**NEW PRIV <priv>**

Definition: .....Specifies the privilege level to be assigned by Opus V1.70 to new files uploaded in this area. This privilege is kept in the files database, and only users with this privilege or higher will be able to "see" the files or download them.

Range: .....Opus privilege level name.

Required: .....No. If not specified Opus will default to the DOWNLOAD PRIV for the file area. If that is not specified, then Opus will default to the AREA PRIV for the file area.

Example: .....NEW PRIV Disgrace

**ORIGIN <string>**

Definition: .....Specifies the Origin line string Opus V1.70 should use for this echomail message area. Opus will automatically append your Net address, you do NOT specify it. An ORIGIN file in the relevant echomail message area will override this setting.

Range: .....Up to 50 printable ASCII characters.

Required: .....No, defaults to the SYSTEM NAME string.

Example: .....ORIGIN Sentry - Longest running FidoNet Node in OZ

**PASSTHROUGH**

Definition: .....Defines an echo area as an area through which messages pass on their way to other nodes in the SCAN list without being stored on the system.

Required: .....No



**PATH <path>**

Definition: .....Specifies the directory in which Opus V1.70 stores the messages for this message area definition.

Range: .....DOS path specification.

Example: .....PATH c:\fido\answers\

**PEEK LOCK <locks>**

Definition: .....Specifies the keys Opus V1.70 requires a user to have before allowing the user to read PRIVATE messages that are not To: or From: them in this message area. Should be useful in "public" echoes that seem to get a lot of private traffic.

Range: .....Opus key/lock specification.

Required: .....No

Example: .....PEEK LOCK DGIUO

**PEEK PRIV <priv>**

Definition: .....Specifies the privilege level Opus V1.70 requires a user to have before allowing the user to read PRIVATE messages that are not To: or From: them in this message area. Should be useful in "public" echoes that seem to get a lot of private traffic.

Range: .....Opus privilege level name.

Required: .....No, if not specified defaults to Sysop.

Example: .....PEEK PRIV Privil

**PICTURE <filename>**

Definition: .....Specifies the file that Opus V1.70 will show in the relevant message or file area to users, who have ASCII menus turned on, before the normal Opus message or file menu. The normal Opus menu will follow.

Range: .....DOS filename specification.

Required: .....No

Example: .....PICTURE c:\opus\menu\piccy

**PREFILES <filename>**

Definition: .....Specifies the file that Opus V1.70 will show users when they choose the FILES option is chosen from the file menu. This file will be shown before the file listing starts.

Range: .....DOS filename specification.

Required: .....No

Example: .....PREFILES c:\opus\files\prefile1



### PRIVATE MESSAGES ONLY

Definition: .....Specifies that all messages created in this message area will be forced private.

Required: ..... Yes, if you wish for all messages in this area to be forced private.

### PUBLIC MESSAGES ONLY

Definition: .....Specifies that all messages created in this message area will be forced public.

Required: ..... Yes, if you wish for all messages in this area to be forced public. This attribute is recommended for all public ECHOMAIL areas.

### PUPLOAD <path>

Definition: .....Specifies the DOS path where files uploaded for this file area will be stored by Opus V1.70.

- UPLOAD The "normal" default upload path.
- PUPLOAD The upload path for users with a privilege level greater than that specified in the PUPLOAD PRIV option.
- KUPLOAD The upload path for users who have the keys specified by the KUPLOAD LOCK option.

Range: .....DOS path specification.

Required: .....  
UPLOAD Yes, if a file area is defined. The path can be identical to the DOWNLOAD path or any other legal DOS path on the system.  
PUPLOAD No  
KUPLOAD No

Examples: .....  
UPLOAD c:\opus\files\games\  
PUPLOAD c:\opus\files\private\  
KUPLOAD c:\opus\files\locked\

### PUPLOAD AREA <number>

Definition: .....Specifies the file area number of the file area that uploads will be considered to be a part of in the Opus V1.70 files database.

Range: .....0-32766, the NUMBER SHOULD MATCH THE FILE AREA NUMBER OF THE FILE AREA WHERE UPLOADS, PUPLOADS or KUPLOADS ARE STORED or I confidently predict that you will be in for some interesting times with your files database.

Required: ..... YES, ESSENTIAL for each upload, private upload or keyed upload path defined!

Examples: .....  
UPLOAD AREA 31 PUPLOAD AREA 41 KUPLOAD AREA 51

**RATIO PRIV <priv>**

Definition: .....Specifies that Opus V1.70 will enforce upload:download ratios in this file area if the user's privilege level is equal to or BELOW that which is specified here.

Range: .....Opus privilege level name.

Required: .....No. If you do not specify a RATIO PRIV in each file area, Opus will default to enforcing ratios in that area.

Example: .....RATIO PRIV Normal

**READ-ONLY**

Definition: .....Specifies that messages cannot be entered in this message area.

Required: .....Yes, if you want to have an area that users can read, but not create messages.

**ROUTE**

Definition: .....Specifies that Opus V1.70 is to route private messages in this echomail message area back to the sender via NetMail instead in the echo. Opus will use the default matrix attributes unless the handling menu has been called to change the attributes.

Required: .....No

**RULES <filename>**

Definition: .....Specifies that Opus V1.70 will show users this file if they have never before been in the file area. The idea behind this is that you might want to have some areas that have minimal restrictions on them, like an area that has nude GIF files in it. You could ask "Are you over 18?" and if the answer is no, send them somewhere else.

NOTE: .....The display of the RULES.BBS file is automatic in message areas if the file exists in the relevant message area directory.

Range: .....Opus text file specification.

Required: .....No

Example: .....RULES c:\opus\misc\nuderule

**SCAN <net/node ...>**

Definition: .....Specifies the network addresses of the systems to which messages from this message area are to be sent.

Range: .....net/node specified as 1/1 through 32767/32767. Opus V1.70 will scan to up to 255 addresses per area.

Required: .....Yes, if this is an ECHOMAIL message area and echomail is to be scanned from this area for sending to one or more nodes. You must specify INBOUND ONLY or SCAN for an ECHOMAIL area.

Example: .....SCAN 106/114 167/99 119/5 141/491

**SECTION <section>**

Description: .....Specifies the Opus V1.70 section(s) to which this area belongs.

Range: .....Opus section specification.

Required: .....No

Example: .....SECTION DGIOU

**TITLE <description>**

Definition: .....Description that is shown to a user by Opus V1.70 when accessing the relevant message or file area.

Range: .....Up to 45 printable ASCII characters.

Required: .....Yes, if message or file area is defined.

Examples: .....TITLE General Messages TITLE Miscellaneous Files

**UPLOAD <path\>**

Definition: .....Specifies the DOS path where files uploaded for this file area will be stored by Opus V1.70.

- UPLOAD The "normal" default upload path.
- PUPLOAD The upload path for users with a privilege level greater than that specified in the PUPLOAD PRIV option.
- KUPLOAD The upload path for users who have the keys specified by the KUPLOAD LOCK option.

Range: .....DOS path specification.

Required: .....UPLOAD Yes, if a file area is defined. The path can be identical to the DOWNLOAD path or any other legal DOS path on the system.  
PUPLOAD No  
KUPLOAD No

Examples: .....UPLOAD c:\opus\files\games\  
PUPLOAD c:\opus\files\private\  
KUPLOAD c:\opus\files\locked\

**UPLOAD AREA <number>**

Definition: .....Specifies the file area number of the file area that uploads will be considered to be a part of in the Opus V1.70 files database.

Range: .....0-32766, the NUMBER SHOULD MATCH THE FILE AREA NUMBER OF THE FILE AREA WHERE UPLOADS, PUPLOADS or KUPLOADS ARE STORED or I confidently predict that you will be in for some interesting times with your files database.

Required: .....YES, ESSENTIAL for each upload, private upload or keyed upload path defined!

Examples: .....UPLOAD AREA 31 PUPLOAD AREA 41 KUPLOAD AREA 51

### UPLOAD LOCK <locks>

Definition: .....Specifies the lock settings on user access to the upload command in this message or file area. A user must have the corresponding keys turned on in their Opus user record for access to the command.

Range: .....Opus key/lock specification.

Required: .....No

Example: .....UPLOAD LOCK DGIUO

### UPLOAD PRIV <priv>

Definition: .....Specifies the minimum privilege level Opus V1.70 requires a user to have before allowing the user to upload messages in this message area or files to this file area.

Range: .....Opus privilege level name.

Required: .....Optional for a defined message or file area.

Example: .....UPLOAD PRIV Normal

## 2.8.7 MENU SECTION

The menu section can optionally be included in the control file. The menu file definition is separately described in section 3 of this manual.

**3****OPUS MENU SYSTEM**

Beginning with Opus V1.10, menus are contained in a single file (eg ENGLISH.MNU or, in some cases, ENGLISH.001). Opus V1.70 adds the following new menus:

- six optional "custom" menus;
- the history menu (the configuration/setup menu, which collected too many new options, was split into the configuration/setup menu and the history menu); and
- an optional section menu,
- and does away with the MATRIX/NETMAIL message menu and the ECHOMAIL message menu.

---

**3.1 MENU MAINTENANCE**

Menus can be modified in a text file and compiled with the NACL compiler. Here is the format of a typical menu item as it appears in an Opus control file:

<command> <privilege> <locks> <description>

For example:

\_FILE Normal !F "File section"

It means the file section command is available to every user of Normal privilege and above who has User Key "F" turned on in their record in the Opus user file.

**3.1.1 ACCESS LEVELS & LOCKS**

It is fairly harmless to change access levels on menu items. Almost every system operator will want to do that. You may also enable locks if your security plan requires it.

Feel free to change access levels and to add or delete locks. You can do so with impunity.

**3.1.2 DELETING MENU COMMANDS**

All menus can be defined in your NACL control file, but if there is a command you do not want to use, you can use a text editor to delete the item from the control file. Do not forget to recompile the control file with NACL.



### 3.1.3 CHANGING THE MENU COMMAND TEXT

Please be careful on this. The menu command text seems innocent, but it is far from that.

The first character of the text is used as the actual menu command character.

One of the nice things about Opus and Fido is the "G" always means GOODBYE. Users never have to guess. If they are on an Opus or Fido BBS, they already know how to use it.

If you make changes without a really good reason, you will be making things complicated and confusing for your users on your system and on other systems which they expect to be similar. The rule to follow is NOBLESSE OBLIGE (can be found in most dictionaries).

Here are some rules:

- 1 No menu command text can be more than 19 characters.
- 2 The first character is the menu command character, and it must be one of the following:

A-Z	an uppercase letter
!	exclamation point
?	question mark
+	plus sign
-	minus sign
=	equals sign
*	asterisk
@	at sign
&	ampersand
- 3 It is your responsibility to make sure you do not duplicate command characters on a menu (i.e., using "H" for two menu options). NACL will not notice this and Opus will only execute the first menu command with the duplicate character.
- 4 If the menu command text has more than one word, Opus may use only the first word from time to time, such as in HOTFLASH menus. You should make sure the first word carries some meaning by itself. You might wish to use the underscore to ensure the full description is shown in the HOTFLASH menus. For example, use "Ans\_Questions", instead of "Ans Questions".
- 5 Do not use a parenthesis in the menu command text. Previous versions of Opus did that [e.g., "G)oodbye"]. Opus now handles that sort of thing automatically. If the user seems to support bright/dim character display, Opus will use brightness to highlight the menu command. It will insert a parenthesis for users who are calling with TTY (also known as "old-tymie") equipment.
- 6 Use capital (uppercase) letters for different MENU SECTION options, for example, MAIN, MESSAGES, FILES, CHANGE SETUP, HISTORY and so on to make these selections stand out on Menus.
- 7 Remember that gratuitous changes can only breed ill feelings. Your users will not thank you if you change the menu command text just for the heck of changing it.

### 3.1.4 ADDING NEW COMMANDS

You can add menu commands, but you have to be careful. Menu commands that begin with an underscore are "global" and can usually appear on any menu without worry.

Other menu commands are intended for use in specific menus. For example, menu commands that begin with "MSG\_" are for use in the message section menu. Although you can put these commands in other menus, it is not recommended.

## 3.2 THE MENU FILE

Opus looks for a file in its language subdirectory (also known as "language path") that has a caller's LANGUAGE NAME as its filename and ".MNU", ".001", ".002", etc. as its extension. If the caller is using ENGLISH, the menu files would be called "ENGLISH.MNU", "ENGLISH.001", "ENGLISH.002", etc.

The numbered extensions are strictly optional and only used if you wish to switch menus between various areas.

If a menu file is missing, the system will try to open the file for the system's default language. If that fails, Opus terminates with an ERRORLEVEL of 3 (NOT safe to have batch file recycle).

### 3.2.1 BUILDING A MENU FILE

There are fifteen menus. NACL will not scream if menus are not defined, which means there is a wonderful opportunity to shoot yourself in the foot. Just call an undefined menu! Here are the fifteen menus:

```
Main
Config
History
Sysop
Message
File
Lore
OpEd
Section
Custom1
Custom2
Custom3
Custom4
Custom5
Custom6
```

The menu section is in its own control file section. Here is a rough outline of how the control file menu section looks:

```
MENU SECTION For English
%
MAIN MENU
%
% Main menu commands go here
%
End Menu
%
CONFIG MENU
%
% Config menu commands go here
%
End Menu
%
HISTORY MENU
%
% History menu commands go here
%
End Menu
%
SYSOP MENU
%
% Sysop menu commands go here
%
End Menu
%
MESSAGE MENU
%
% Message menu commands go here
%
End Menu
```

```
%  
FILE MENU  
%  
% File menu commands go here  
%  
End Menu  
%  
OPED Menu  
%  
% OpEd menu commands go here  
%  
End Menu  
%  
  
LORE Menu  
%  
% Lore menu commands go here  
%  
End Menu  
%  
SECTION MENU  
%  
% Section menu commands go here  
%  
End Menu  
%  
CUSTOM1 MENU  
%  
% Custom1 menu commands go here  
%  
End Menu  
%  
CUSTOM2 MENU  
%  
% Custom2 menu commands go here  
%  
End Menu  
%  
CUSTOM3 MENU  
%  
% Custom3 menu commands go here  
%  
End Menu  
%  
CUSTOM4 MENU  
%  
% Custom4 menu commands go here  
%  
End Menu  
%  
CUSTOM5 MENU  
%  
% Custom5 menu commands go here  
%  
End Menu  
%  
CUSTOM6 MENU  
%  
% Custom6 menu commands go here  
%  
End Menu  
%  
END MENU SECTION
```

If your system uses more than one language, you can define all of the menu files using the same control file (if you want to do it that way).

### 3.2.2 MENU PROMPTS

Beginning with Opus V1.70, you can change the default menu prompts. For example:

```
%
CONFIG MENU "User Configuration"
  SET_HELP Disgrace "Help Level"
  SET_NULLS Disgrace "Nulls"
  _SHOW Twit "?HELP" = c:\hlp\Config
END Menu
%
```

When a user is at the prompt for the CONFIG MENU, instead of the default "CHANGE [99]:" the user would get "User Configuration [99]:". This was put in for the CUSTOM menus, but you can use it for all menu prompts except the LOCAL/MATRIX/ECHOMAIL message prompts.

### 3.2.3 MENU COMMANDS: General information

There are several classes of menu commands. The classes roughly coincide with the various menu types. In other words, you will find menu commands that relate to a file section and others that relate to a message section.

You can usually tell the class of a menu command from the first part of its type name:

```
F_????? ..... a file section command
MSG_???? ..... a message/echo/mail section command
SYS_???? ..... a sysop section command
ED_????? ..... an editor menu command
SET_???? ..... a change user configuration/setup command
USR_???? ..... an history menu command
```

Menu commands that have an underscore as the first character are considered "global"... they are not associated with any specific menu type and there are usually no restrictions on their use.

Except for LORE and OpEd commands, you can mix and match menu commands. Putting an editor command into any menu other than an editor menu will cause an error condition. The two editor menus can only contain "ED???" commands or \_SHOW.

Saying you can "mix and match" is not exactly correct. When you change menu types (eg go from the MAIN menu to the MESSAGE menu), the system does some setup work for the new menu type. As long as you stick within the menu commands designated for the current menu type, there will be no further setup work. On the other hand, if you use a command associated with some other menu type then the system will do setup work each time that menu command is selected -- causing the whole session to slow down and doing more wear and tear on your disk drive than would otherwise occur.



### 3.2.4 GLOBAL MENU COMMANDS

Global Menu Commands	Description
<code>_MAIN</code>	Set active menu type to MAIN menu
<code>_SETUP</code>	Set active menu type to CONFIGURATION/SETUP menu
<code>_MESSAGE</code>	Set active menu type to MESSAGE menu (there is only one message menu in V1.70; the EchoMail Message Menu and Matrix/NetMail Message Menu in V1.1x have gone).
<code>_HISTORY</code>	New Opus V1.70 menu command which allows user into the history menu. This is a new menu for V1.70. It was created because the Configuration menu was getting very full History menu options are about the user, Configuration menu options are about how Opus will look to the user. Several of the old "SET_" menu options have been moved to the History menu.
<code>_ECHO</code>	OBSOLETE (In V1.1x set active menu type to EchoMail menu).
<code>_MAIL</code>	OBSOLETE (In V1.1x set active menu type to Matrix/NetMail Message Menu.)
<code>_SYSOP</code>	Set active menu type to SYSOP menu.
<code>_FILE</code>	Set active menu type to FILE menu
<code>_SECTION_MODE</code>	New Opus V1.70 menu command selects the SECTION menu. Can be used anywhere except in the LORE and OpEd menus.
<code>_PICK_SECTION</code>	New Opus V1.70 menu command changes the user from the current Section they are in to a new one.
<code>_MY_SECTION</code>	New Opus V1.70 menu command selects the USER Defined Section as the active Section for the user. See "USER Defined Section" later on. This menu command can be used anywhere except in the LORE and OpEd menus.
<code>_MAKE_SECTION</code>	New Opus V1.70 menu command to allow user to create their own section definition. See "USER Defined Section" later on. This menu command can be used anywhere except in the LORE and OpEd menus.
<code>_SYSOP_MESSAGE</code>	New Opus V1.70 menu command that allows users to enter a message to the Sysop from wherever the command is available. It is a variant of the logoff message in that users are not asked for names or subject or anything, but when they have finished they return to where they were in Opus instead of being disconnected. May be useful as a replacement for Yell?
<code>_SEND_MESSAGE</code>	<p>New Opus V1.70 menu command that allows a user to send a message to a user on another task. Whenever Opus hits a menu it will look for a file in the STATUS directory called CHAT## where ## is this task number in hexadecimal. When the message in CHAT## has been read by the other user, Opus will delete the file.</p> <p>When the user chooses this command, Opus will look in the STATUS directory for ACTIVE##.DAT files. For each file that it finds it will read the corresponding LASTUS##.DAT file to see who is online.</p> <p>If the user in Lastus##.DAT has BLOCK CHAT turned on, Opus will ignore that user unless the person who initiated the chat is Sysop privilege. Otherwise, Opus will list the task number the user is on, their name and city (respecting the user list options). The user initiating the chat can then choose which user to send a message to by selecting the relevant line number and then entering as many lines of message text as appropriate.</p>
<code>_SEE_CHAT</code>	New Opus V1.70 command that polls the status directory to see if there is a chat-type message waiting for the user without the user having to change menus or hit enter at the



	menu. If there is no appropriate CHAT## file, Opus will report back "Nothing for you". (Refer to <code>_SEND_MESSAGE</code> )
<code>_SEE_LINES</code>	New Opus V1.70 menu command to see who else is online in multi-line systems using a common status directory. (Refer to <code>_SEND_MESSAGE</code> )
<code>_MAKE_MSG_MACRO</code> <code>_MAKE_FILE_MACRO</code> <code>_MAKE_MACRO</code>	New Opus V1.70 menu commands which store up to 15 characters for a user keyboard macro in one of three macro fields in the user record. When user presses the macro key, whatever has been stored in the user record for that macro will be executed as if the user had typed in the whole thing. There are three macro fields in the user record. A Message Macro, a File Macro, and a Default Macro. If a user is at a message menu when hitting the macro key, the Message Macro will be executed or changed.
<code>_MY_MACRO</code>	New Opus V1.70 menu command which executes whatever has been stored in the user's macro field for the current menu. If at the Message menu, Opus will execute the Message Macro. If at the Files menu, Opus will execute the File menu. Anywhere else, Opus will execute the Default menu.
<code>_GOODBYE</code>	Asks user to confirm his/her intentions, If there is an "area zero" message area, asks if caller wants to leave a comment message (Opus does not ask for the TO: or SUBJECT fields, but automatically fills in the sysop's name and "Logoff comment from <username>".
<code>_STATS</code>	Displays chart of such things as time on- line.
<code>_SHOW &lt;filename&gt;</code>	Executes the specified *.BBS or *.GBS file; it will not display a file with any other file extension.
<code>_YELL</code>	If currently in a <code>_YELL</code> event, makes noise on sysop's computer to page him/her for a chat.
<code>_USERLIST</code>	Displays a list of ALL users or all users whose names match a given partial name (if you do not want certain users to be listed, turn off the option in the users' user records with OUFM or some other user file editor).
<code>_VERSION</code>	Displays details about Opus (version number, credits etc) and its environment (eg, Computer class, Operating System, and fossil driver).
<code>_OUTSIDE = DOS &lt;program&gt;</code>	Execute an external program loading a copy of COMMAND.COM. On exit to Opus, the LASTUSER file is re-read (see also <code>_OUTSIDE = SDOS</code> ). This method should be used for calling batch files from Opus.
<code>_OUTSIDE = SDOS &lt;program&gt;</code>	New Opus V1.70 command to execute an external program loading a copy of COMMAND.COM. On exit to Opus, the LASTUSER file is NOT re-read regardless of the SECURE setting (see also <code>_OUTSIDE = DOS</code> ). This method should be used for calling batch files from Opus.
<code>_OUTSIDE = RUN &lt;program&gt;</code>	Execute an external program as a child process. On exit to Opus, the LASTUSER file is re-read (see also <code>_OUTSIDE = RUN</code> ).
<code>_OUTSIDE = SRUN &lt;program&gt;</code>	New Opus V1.70 command to execute an external program as a child process. On exit to Opus, the LASTUSER file is NOT re-read regardless of the SECURE setting (see also <code>_OUTSIDE = SRUN</code> ).
<code>_OUTSIDE = EXIT &lt;number&gt;</code>	Cause Opus to exit with the specified DOS errorlevel.

### 3.2.5 CONFIGURATION/SETUP MENU COMMANDS

Config Menu Commands	Description
SET_MENU	New Opus V1.70 menu command that allows users to always return to their last menu after logging on. AFTER going through the usual logon procedure. The only difference is that instead of the MAIN menu being the menu where users are dumped, they will be dumped at the last menu they logged off last time.
SET_ASKGRAPH	New Opus V1.70 command that asks the user what type of graphics they want on each call. It was requested for people who call from a variety of terminal types during the day.
SET_GMENSUS	New Opus V1.70 menu command for ANSI/graphic menus. Users get long, tedious graphic menus BEFORE the usual Opus menus. (Ideally a user would set themselves to Hotflash or Expert help with this command.) The actual OEC files that are displayed are stored in the PRM file as CONFMENU, MAINMENU, SECTMENU, and SYSOPMENU and as PICTURE in each file area where you want to use them. Use full path/file names without extension. The usual *.BBS and *.GBS standards are used. This OEC might just be extended help for the menu choices you have, or it might be an OEC menu that can be combined with the new ^OG OEC.
SET_BLOCK	New Opus V1.70 menu command to allow users to block interline chat. Tells other users that the user does not want to be interrupted by them on a multiline system.
SET_TIME	New Opus V1.70 menu command to show the time remaining at every prompt if the user turns this on.
SET_ASKGRAPH	New Opus V1.70 menu command to ask users to select what type of graphics command to use at every logon, right after giving their password.
SET_KEYS	New Opus V1.70 menu command to allow users to have Hotkeys at all help levels (ie, users do not have to press ENTER after each command).
SET_PWD	OBSOLETE (change user password in V1.1x).
SET_HELP	Change user help level.
SET_NULLS	Change whether Opus sends nulls (and if so, how many) or not.
SET_LEN	Change user screen length.
SET_WIDTH	Change user screen width.
SET_TABS	Change whether Opus sends tab characters or equivalent numbers of space characters.
SET_MORE	Enable or disable "More?" prompting.
SET_VID	Change remote video mode (Avatar, ANSI, ASCII).
SET_CLS	Change whether to send clear screen codes or carriage returns to the user's terminal to clear the screen.
SET_DEF	<Gasp!> New Opus V1.70 menu command to set a default file transfer protocol. If one has been set by the user, Opus will use it and not ask the user for a protocol when the upload and download options are chosen.  *** WARNING *** This will mess up the automatic Zmodem "R)Zup-load" menu option that a lot of grizzled veterans provide for their users.

SET_EDIT	Select editor (OpEd or LORE).
SET_IBM	Select IBM or "equivalent" ASCII character set.
SET_CITY	OBSOLETE (In Opus V1.70, see USR_CITY.)
SET_LANG	OBSOLETE (In Opus V1.70, see USR_LANG.)
SET_LIST	OBSOLETE (In Opus V1.70, see USR_LIST.)

### 3.2.6 HISTORY MENU COMMANDS

History Menu Commands	Description
USR_NAME	New Opus V1.70 menu command to allow users to edit their logon name.  WARNING: Opus cannot re-index USER.NDX internally. After a user changes their name they can continue to logon just fine, but Opus will have to do a linear search of USER.DAT for them. If you are going to allow users to change their names, run USERNDX.EXE periodically to put them back into the index file. Opus will have to do a linear search of USER.DAT for users who have changed names until you do.
USR_ALIAS	New Opus V1.70 menu command to allow users to change their alias. It will check to make sure that they are not trying to set their alias to the real name of some other user in your USER.DAT file. This also extends to new user logon. Users could have logged on and claimed any alias, now Opus will always check to make sure that the name is not already being used.
USR_PWD	Allows users to change their passwords (SET_PWD in V1.1x).
USR_CITY	Allows users to change their city (SET_CITY in V1.1x).
USR_ADDRESS	New Opus V1.70 menu command to allow users to edit their street address.
USR_PHONE	New Opus V1.70 menu command to allow users to change their phone number field.
USR_BIRTHDAY	New Opus V1.70 menu command to allow users to edit their birth date.
USR_ANSWER	New Opus V1.70 menu command to allow users to edit the answer they gave in answer to LOGON MyQuestion. See "LOGON MYQUESTION" in the Session Section for more information.
USR_LIST	Allows users to change the way they are listed in the user list (SET_LIST in 1.1x).
USR_LANG	Allows users to change their preferred language (SET_LANG in 1.1x).



### 3.2.7 MESSAGE MENU COMMANDS

Message Menu Commands	Description
MSG_QUICK	New Opus V1.70 menu command which searches forward in the current message area from the next message to the last message looking for messages that are TO: the current user. When it finds one, it stops and displays the message.
MSG_QINQ	New Opus V1.70 menu command which searches forward from the current message looking for a match to a string that the user provides. When it finds a match, Opus will display the message. This command remembers the last search string that a user asked for, so hitting ENTER will allow a user to continue a search without having to type in the string for each match.
MSG_NAREA	New Opus V1.70 menu command that changes to the next area that has new (ie unread) messages in it up to the high message area you've set in your CTL file. Should be real handy for people who write fancy communications scripts. If Opus does not find anything it will just stay in the current area.
MSG_HELP = <filename>	<p>New Opus V1.70 menu command that works just like _SHOW does, except that if there is a help file defined in the message area it overrides the menu command. Think of the MSG_HELP as being a default help file that can be changed according to areas. For example:</p> <p>MSG_HELP Disgrace "?Help" =C:\Help\Msg HELP C:\Help\Matrix (control file extract)</p> <ul style="list-style-type: none"> <li>• If the user is in the message area that has the line "HELP C:\Help\Matrix" in its configuration in the control file, and the user hits "?" the user will see C:\Help\Matrix.</li> <li>• If the user is in any other area, the user will see C:\Help\Msg.</li> </ul>
MSG_RULES	New Opus V1.70 menu command to make it possible for users to review the rules at any time, even after reading messages. It is a "smart" menu item in that when the message area is initialized Opus will go check to see if RULES.BBS or RULES~.BBS exist in the message path and only include it in the menu if such a file exists.
MSG_SCAN	Scans for new messages, but in Opus V1.70 it only works if a user is in a Section.
CHG_AREA	Allows user to select a new message area.
MSG_NEXT	Set message direction to FORWARD and read next message.
MSG_PRIOR	Set message direction to BACKWARD and read next message.
MSG_LAST	Redisplay current message.
MSG_EDIT_NEW	In local mode, call local editor, if one was declared. Otherwise, call local editor, OpEd or LORE as defined by SET_EDIT.
MSG_EDIT_REPLY	In local mode, call local editor, if one was declared (Opus will quote the message and place the original writers initials in the reply automatically). Otherwise, call local editor, OpEd or LORE as defined by SET_EDIT.
MSG_NONSTOP	Begin non-stop display of messages in current direction.
MSG_CHILD	Display reply to current message, if one exists.
MSG_PARENT	Display parent to current message, if this is a reply.
MSG_LIST	Display list of message TO:, FROM: portions of header. If user specifies verbose mode, also display Subject lines.

MSG_INQ	Searches for specified string in TO:, FROM:, and SUBJECT: fields.
MSG_KILL	Delete a message.
MSG_HURL	Move a message to another area.
MSG_FRWD	Forward a message to another user.
MSG_UPLD_NEW	Upload message via Xmodem protocol.
MSG_UPLD_REPLY	Not implemented.
MSG_XPRT	Export a message to ASCII text file.

### 3.2.8 SYSOP MENU COMMANDS

Sysop Menu Commands	Description
SYS_AREA	OBSOLETE (sysop area maintenance).
SYS_MAIL	Display sysop matrix information and menu.



### 3.2.9 FILE MENU COMMANDS

File Menu Commands	Description
F Locate	Locate a file (crosses file area boundaries).
F TITLES	Display filenames and descriptions for this area from the files database or find <string> in this file area.
F LONG	New Opus V1.70 menu command that will show a two-line listing instead of the single line listing. The two lines includes date of upload, how many times downloaded, and optionally who uploaded the file. You might want to replace the short list, or allow it as a second option. The "WHO PRIV" in the control file will be respected, but users could find out the date a file was uploaded and how many times its been downloaded.
F NEW	New Opus V1.70 menu command that searches the entire files database for files that have been added/uploaded since the user's last logon. The user can override this and set a new number of days to search for the search period. If the user has already logged on today, Opus will default to 14 days.
F DSPL	Display a text file to screen from the current file area.
F DNLD	Download a file to user.
F DNHANG	New Opus V1.70 menu command which allows users to start a file download and, if Opus thinks the download is successful, Opus will give the user 10 seconds to hit any key or it will automatically hang up the modem.
F UPLD	Upload a file to Opus.
F RAWDIR	Display DOS directory of current file area directory.
F ARCDIR	Show list of files in specified compressed file in this area (handles ARC, ZIP, LZH, LHA, PAK, ARJ compression).
F KILL	Delete file in current area and mark its entry in FILESBBBS.DAT as deleted.
F HURL	Move a file from one subdirectory to another. Moves across physical disk drives ARE supported in Opus V1.70.
CHG AREA	Allows user to select a new file area.
F TEMP	New Opus V1.70 menu command that allows user/sysop to set a temporary upload/download path and work from it. The temporary path will disappear as soon as they change areas or do anything that causes an area change. The temporary upload/download directory can be any valid directory on any valid drive. Opus will not ask for descriptions for files uploaded in the temporary directory, they will not be included in FILESBBBS.DAT files database but you can use the Upload, Download, Contents, Raw, and Type commands in the temporary directory without any problems. As soon as you change areas or try to do a locate the normal paths will take over. Uploads and downloads in the temporary path will not be included in the upload or download logs but will be in the main log file.
F ADOPT	New Opus V1.70 menu command that allows you to add files to FILESBBBS.DAT from the keyboard. You select the F ADOPT menu choice and Opus will ask you for a file name. Then it will confirm that you want to add the file. (Opus will NOT check to see if the file already exists in FILESBBBS.DAT, only that it exists in the download directory.) Then you'll be asked for a description. The person who adopts the file is considered to be the uploader. You can use this command combined with the F TEMP command to adopt a file that isn't in the download path for the current area. Use F TEMP

first to set the path to use, then F\_ADOPT to adopt the file(s). Yes, you can use wild-cards.

F\_HELP = <filename>

New Opus V1.70 menu command that works just like \_SHOW does, except that if there is a help file defined in the file area in the control file, then that help file overrides the menu option. Think of the F\_HELP file as being a default help file that can be changed according to areas. For example:

F\_HELP Disgrace "?Help" = C:\Help\File1 HELP C:\Help\Files (control file extract)

- If the user is in the file area that has the line "HELP C:\Help\Files" in its configuration in the control file, and the user hits "?" the user will see the file C:\Help\Files.
- If the user is in any other area, the user will see the file C:\Help\File1.

### 3.2.10 EDITOR MENU COMMANDS

Editor Menu Commands	Description
ED_SAVE	LORE Save current message in editor.
ED_ABORT	LORE Discard current message in editor.
ED_LIST	LORE Display current message in editor.
ED_CHG	LORE Change a line in current message in editor.
ED_INSRT	LORE Insert a line in current message in editor.
ED_DEL	LORE Delete a line in current message in editor.
ED_CONT	LORE Continue adding text to message in editor.
ED_DISK	LORE Opus will ask for a filename. If OpEd you have started editing a message, it is lost, only the file you read in is saved.
ED_TO	LORE Change TO part of header in OpEd current message in editor.
ED_FROM	LORE Change FROM part of header in OpEd current message in editor.
ED_SUBJ	LORE Change SUBJECT part of header in OpEd current message in editor.
ED_ATTR	LORE Change HANDLING attributes in OpEd header of current message in editor.
ED_UUCP	LORE New Opus V1.70 command to turn on OpEd or off the automatic insertion of the UUCP address string (if it exists) in replies when they are saved by Opus.



## 4

# OPUS EMBEDDED COMMANDS

Opus Embedded Commands (OEC) are very powerful and can be used to great advantage to highly customize your Opus.

The Opus Embedded Commands system has been extensively rewritten by Doug for Opus 1.70. The number of OEC files (ie, \*.BBS or \*.GBS) you could run in Opus 1.14 was 10, after which Opus dumped the user back to a menu. In Opus 1.70 there is NO LIMIT! Opus 1.70 also enables you to stack OEC files, where one file calls another and then returns to the original point.

This section is intended to serve as a reference to all available Opus Embedded Codes (OECs). Examples, and discussion of their use, are included in the Opus 1.70 Sysop Operations Manual.

---

### 4.1 TASK NUMBER SUPPORT

Embedded codes may be used to display different files according to which task number your Opus is using. This is possible by use of the '#' character in the filename or path reference. Only tasks 0-9 are supported, even though the Opus task number can be specified as a higher number. Remember that DOS filenames cannot be longer than 8 characters. Your task selected filenames must be limited to 7 characters plus one character for the task number. For example,

```
^OSc:\opus\misc\help#
```

would show:

```
c:\opus\misc\help1.bbs/gbs for task 1.  
c:\opus\misc\help2.bbs/gbs for task 2.
```

Starting with Opus 1.70, the '#' option also applies to external program command line parameters used with the ^OC, ^OH, ^OJ, ^OK, ^OW, ^O2 and ^O3 Opus Embedded Commands. For example,

```
^OCc:\opus\update.exe today#.dat  
^OHc:\opus\update.exe today#.dat
```

would run:

```
^OCc:\opus\update.exe today1.dat for task 1  
^OCc:\opus\update.exe today2.dat for task 2
```

NOTE: Task number substitution will NOT work for the LOGO.BBS file which Opus transmits as soon as a caller or mailer connects.

---

### 4.2 LANGUAGE SUPPORT

Embedded codes may be used to display different files according to the user language number selected. Opus will substitute the HEX number of the language of the current user for '~' in the file name or path.

There are two important exceptions:

AREAINF~.BBS is hardcoded, but goes to AREAINFO.BBS if there is no language match.

WHY~????.BBS is hardcoded, but goes to WHY\_????.BBS if there is no language match.

Other than that, you can control any filename/path, so you can insert the '~' wherever you want.



Remember that DOS filenames cannot be longer than 8 characters. Your language selected directory and filenames must be limited 7 characters plus one character for the language number. For example:

`^OSc:\opus\languag~\myfile`

would expand to show:

`c:\opus\languag1\myfile.bbs/gbs` for user language 1   `c:\opus\languag2\myfile.bbs/gbs` for user language 2  
`c:\opus\languagA\myfile.bbs/gbs` for user language 10   `c:\opus\languagC\myfile.bbs/gbs` for user language 12

Also applies to external program command line parameters used with the ^OC, ^OH, ^OJ, ^OK, ^OW, ^O2 and ^O3 Opus Embedded Commands.

---

## 4.3 OEC (BBS/GBS) FILES

An Opus OEC file comes in two file extension flavours: BBS and GBS. The GBS file extension is really a hangover from days gone by; files with the GBS extension are only displayed to users who have either ANSI or Avatar video modes selected. If you have ANSI codes in a GBS file, and the user has selected the Avatar video option, the user will get lots of raw ANSI codes transmitted (ie not a good idea).

An Opus OEC file with the BBS file extension will handle all user video modes (ie no graphics; ANSI graphics; and Avatar graphics) with the one file. You can include colour commands, cursor positioning, etc. There is, of course, one gotcha! YOU MUST USE AVATAR CODES AND NOT ANSI CODES in the BBS file.

Opus will use the video configuration selected by each user to determine how to display the BBS file. If the user has Avatar graphics turned on, then Opus will transmit the Avatar codes to the user; if the user has ANSI graphics turned on, then Opus will convert the Avatar codes into the appropriate ANSI codes and transmit them to the user; if the user has no graphics enabled, then Opus will strip the Avatar codes and transmit only the plain text to the user. Opus will even check to see if the user has IBM graphics characters enabled and, if not, will translate the box drawing symbols and the like to the equivalent ASCII characters.



## 4.4 EMBEDDED COMMANDS DICTIONARY

The following sections list all embedded codes supported by Opus 1.70. These listings are intended for reference. Please refer to the Opus 1.70 Sysop Operations Manual for examples and further explanation on usage.

NOTES: Some command characters in this section are CONTROL characters. The '^' symbol means 'control'. For example, '^F' represents CONTROL-F.

All OECs Should be in UPPER CASE.

### 4.4.1 MISCELLANEOUS COMMANDS

Command	Description
^A	'Press ENTER to continue'.
^B	Disable break ( K but not O).
^C	Allow break ( K but not O).
^D	If within 5 lines of end-of-screen, show a 'More?'.
^E	Enable 'More?'.
^F	See F codes.
^G	Ring the caller's bell.
^H	Backspace.
^I	Tab.
^J	MS-DOS Carriage Return/Line Feed pair.
^K	Turn 'More?' off.
^L	Clear screen.
^O	See O Codes.
^P	See Privilege Codes.
^V	See Avatar Codes.
^X	See Expiration Codes.
^YC#	Replicate C # times (# is an ASCII character and the number of times it represents is equal to that character's ASCII value, so a CR is equal to 13 times, and P is equal to 80 times while p is equal to 112 times).

#### 4.4.2 AVATAR COMMANDS

The video mode option in the Opus CONFIG menu section allows the commands to be transmitted directly to a user for a significant increase in screen painting speed.

Command	VT-100 sequence	Description
^V^A^A	<esc>[0;34;40m	Blue
^V^A^B	<esc>[0;32;40m	Green
^V^A^C	<esc>[0;36;40m	Cyan
^V^A^D	<esc>[0;31;40m	Red
^V^A^E	<esc>[0;35;40m	Magenta
^V^A^F	<esc>[0;33;40m	Brown
^V^A^G	<esc>[0;37;40m	LightGrey
^V^A^H	<esc>[0;1;30;40m	Grey
^V^A^I	<esc>[0;1;34;40m	LightBlue
^V^A^J	<esc>[0;1;32;40m	LightGreen
^V^A^K	<esc>[0;1;36;40m	LightCyan
^V^A^L	<esc>[0;1;31;40m	LightRed
^V^A^M	<esc>[0;1;35;40m	LightMagenta
^V^A^N	<esc>[0;1;33;40m	Yellow
^V^A^O	<esc>[0;1;37;40m	White
^V^B	<esc>[5m	Blinking on. Blinking is turned off by the next 'V A' command
^V^C	<esc>[1A	Move cursor up one line
^V^D	<esc>[1B	Move cursor down one line
^V^E	<esc>[1D	Move cursor left
^V^F	<esc>[1C	Move cursor right
^V^G	<esc>[K	Clear to end of line
^V^H <R> <C>	<esc>[##;##f	Set cursor position. The ##'s are 1-based numbers with the 'Y' (row) coordinate first. The upper left of the display is '1,1.' The start of the second line would be '2,1.' In most cases, the lower right corner will be '23,80.' Values that fall outside of the user's declared screen size are CLIPPED.
^L	<esc>[H <esc><esc>[0;30;36m <esc>[2J <esc>[J	Home and clear screen. Yes, all of those VT-100 commands are generated for a clear screen. It is the only sequence we have found that does the job on all VT-100 and ANSI displays.

## 4.4.3 ^F COMMANDS

Command	Description
^FA	Display the total number of messages a user has read.
^FB	Display the total number of messages entered by user.
^FC	Does the _SEND_MESSAGE menu command for inter-line chat.
^FW	Displays who else is online by checking the ACTIVE##.DAT and LASTUS##.DAT files in the Opus status directory.
^FL	Send a message to the user on the current Opus task (intended for use in BYEBYEBBS).
^FR	Looks for the CHAT## file in the Opus status directory and displays it if it exists.
^FM	Displays which menu is active. Gives the appearance of being at the relevant menu prompt inside Opus.
^FS	Initiates a the same function as the _SYSOP_MESSAGE menu command. This terminates an OEC file.
^FW	Displays who else is online on multiline systems.
^FZ	Do a memory dump in the Opus log file (for debugging only).
^F^A	Display quote of the moment.
^F^B	Display user's name.
^F^C	Display user's city/state.
^F^D	Display today's date.
^F^E	Display the total number of calls user has made to system expressed as an ordinal number (*eg. '1st', '2nd', etc).
^F^F	Display the user's first name.
^F^G	Dramatic one-second pause.
^F^H	Displays the last time/date that a user accessed the current file area.
^F^K	Displays the user's total minutes on-line in the last 24- hours, including the time for the current call.
^F^L	Displays the length of the user's current call so far (in minutes).
^F^N	Control-niel (disconnect) works remotely and in keyboard mode.
^F^O	Displays the number of minutes remaining for this call.
^F^P	Displays the date/time when the user has to be off the system.
^F^Q	Displays the total number of user calls to Opus to date (ordinal number).
^F^R	Displays the user's NET downloads today (downloads minus uploads). If uploads are greater than downloads, this number will be negative.
^F^T	Displays the current time.
^F^U	On a questionnaire, all answers are required.
^F^V	On a questionnaire, all answers are optional.
^F^W	Displays the total bytes uploaded by the user.
^F^X	Displays the total bytes downloaded by the user.
^F^Y	Displays the user's download:upload ratio.

NOTE: Several ^F commands deal with the amount of time a user has remaining. Until the user is totally through the login procedure, he/she will have about 10 minutes (changeable in the control file setting LOGIN <Timelimit>). The use of these commands prior to the first MAIN MENU may produce unexpected results.

## 4.4.4 ^O COMMANDS

Command	Description
^OA<file>	Shows the specified OEC file <file> and returns to the current location in the original file. You can stack this command. It is a doubly linked list so the limitation is how much memory you can find, but Doug recommends keeping it down to 4-5 deep (see also OS).
^OB<baud>	Shows the rest of the line only if user's baud is greater than or equal to <baud>.
^OC<prog>	Call/execute specified program (system()), and re-read LASTUS##.DAT. Stuffs the command buffer with ENTER before returning to Opus (see also OH, OJ, OK, O2, O3).
^OD	Set the default directory for GBS/BBS OEC files.
^OE	Skip if not full-screen.
^OF<file>	On exit from the current OEC file, show the specified OEC <file> next.
^OG<cmds>	Stuffs <cmds> into the Opus command buffer and terminates the OEC file. Valid commands are any which are available from the CURRENT menu. You can also use the HEX menu codes by preceding them with an escape character.
^OH<prog>	Call/execute the specified program (system()), and do not re-read LASTUS##.DAT no matter what SECURE is set to. Stuffs the command buffer with ENTER before returning to Opus (see also OC, OJ, OK).
^OI	Gets and saves a string of up to 58 characters from the user. When Opus hits a OC, OH, OJ, OK, O2, O3 or OG, whatever the user has entered will be appended to the command given. Opus will insert a space between the command that you give and the string the user types.
^OJ<prog>	Like OC, does a system() call, but does not stuff ENTER in the command buffer (see also OC, OH, OK, O2, O3).
^OK<prog>	Like OH, does a system() call, but does not re-read LASTUS##.DAT no matter what SECURE is set to, but does not stuff ENTER in the command buffer (see also OC, OH, OJ, O2, O3).
^OL~@@	If user language is equal to , jump to byte @@, where is the hexadecimal language number 1-9 or A-C and @@ is a decimal number.
^OM<str>	Store user's multiple choice response to answer file. <str> is an optional string which may be used to identify the question.
^ON<str>	Store user-supplied line to answer file. <str> is an optional string which may be used to identify the question.
^OO<file>	Open specified answer file for storing user responses.
^OP	Post user stats to answer file.
^OQ	Quit the current OEC file immediately.
^OR<list>	Wait for character to be input by user, then check for match in the specified multiple choice <list> of characters. If no match, prompt user to "Try again.". If there is a match, continue processing the OEC file (used in conjunction with OU). The   "pipe" character is used to represent ENTER.
^OS<file>	Show the specified file (see also OA).
^OT	Go to the top of current OEC file.
^OU<char>	User response conditional. Show rest of line if the character input from the user (see OR) matches <char>, otherwise continue processing the OEC file.
^OV###	Go to byte ### in this file.
^OW<file>	Checks for the existence of <file>. If the file does not exist Opus will skip the rest of the line. Wildcards are allowed, and the substitution for language and task numbers is honoured.
^OX<pos>	OX records the current position in the current file, jumps to <pos> and when OY is encountered, Opus returns. This command can be stacked up to 10 levels deep.
^OY	Return from OEC file now.
^OZ<err>	Compares the DOS errorlevel returned from the O2 and O3 commands and operates just like "if errorlevel" does in DOS batch files.
^O0###	Go to byte ### in this file.
^O2<prog>	Spawns <prog>. Spawnlp() is different from system() because it returns the DOS errorlevel of the program that was called. System() only reports success or failure. Use this for OC if you want to do something with the errorlevel (see also OC, OH, OJ, OK, O3).
^O3<prog>	This is the spawnlp() equivalent of the OH command. Does a spawnlp() and runs the specified program but does not re-read LASTUS##.DAT no matter what SECURE is set to (see also OC, OH, OJ, OK, O2).



#### 4.4.5 PRIVILEGE AND KEY COMMANDS

Command	Description
^PA<keys>	Add <keys> for the user.
^PC<keys>	Clear <keys> for the user.
^PB@	If user priv is below @ (first letter of privilege level), skip rest of line.
^PF@	If user priv is below @, end file now.
^PH<hlp>	If the user's help level is not equal to the help level on the line, the rest of the line is skipped. <hlp> is one of: N for Novice, R for Regular, E for Expert or H for Hotflash).
^PK<keys>	If user does NOT have keys <keys>, end file now.
^PL@	If user priv is above @, sees rest of line.
^PN<keys>	If user does NOT have keys <keys>, sees rest of file.
^PO<keys>	If user does NOT have keys <keys>, sees rest of line.
^PP<keys>	If user has keys <keys>, sees rest of line.
^PQ@	If user priv is equal to @, sees rest of line.
^PX@	If user priv does NOT equal @, sees rest of line.
^PY~	If user language is NOT , skip rest of line.
^PZ~	If user language is , skip rest of line.
^PS<sec>	If user section is equal to <sec>, sees rest of line.
^PT<sec>	If user section is NOT equal to <sec>, skip rest of line.

NOTE: To end a key sequence, YOU MUST USE A SPACE CHARACTER. For example, use ^PKABC HOWDY and not ^PKABCHOWDY or Opus will naturally enough think the user needs keys for locks A, B, C, H, O, W, D, and Y.

#### 4.4.6 EXPIRATION COMMANDS

Command	Description
^XC	If user expiration flag is by date, show when access expires as '14 Jul 91'.
^XD	If user expiration flag is by date, show number of days until expires.
^XE	If user expiration flag is by date, show rest of line.
^XM	If user expiration flag is by time, show number of minutes remaining.
^XN	If user expiration flag is by time, show rest of line.

#### 4.4.7 FILES DATABASE OECs

When displaying files, if there's a ^V or <ESC> on the line, Opus will NOT try to wrap the line. It will just show it "as is". This should help people who are doing graphics inside the FILES.BBS.DAT files database.

Opus 1.14 and earlier allowed you to use all the OECs in FILES.BBS. Opus 1.70 will allow you to include almost all of the OECs, except the ^O group, in the new FILES.BBS.DAT files database. Only the ^OQ command of that group will work in files listings from the files database. All the other OECs should work in file listings from the files database.

---

## 4.5 FULL TILT BOOGIE COMMANDS

And now a word from our sponsor:

"[Full Tilt Boogie (FTB) is] an old idea, and it looks like BBS'ing didn't much like it. <sigh>

"The idea was for a 'board' and a 'user' to come closer together. Today we have really formal mail sessions and really formal user sessions. You have one or the other. Either the system is talking on FidoNet or to a human caller.

"With FTB, another system could call and do things like humans do. The other system could get into a message area, set MEADOW as the current area, and send a command to mean 'send all messages received after such-and-such a date.' The material would be compressed on the fly.

"The same kind of thing would work for files. H'mmm, I guess you could extend that to include external programs.

"There would be nothing keeping human callers with relatively snazzy software from using the same mechanism because FTB would actually be a 'help level'. What we call 'help level' is really an 'interface mode' internally to Opus.

"That would eliminate EchoMail processing off-line. It would make the receiving system the driving force in EchoMail, and the sending system wouldn't have to care who had gotten which messages. If every system in the network (or in one conference) were FTB, it would eliminate the need for SEEN BY lines ... and that accounts for between 40 and 60 percent of each message (read 'cheaper to transmit' here).

"As a sysop, you would have the same control over a calling system as you do over calling humans. There would be some kind of initial privs, etc.

"That's FTB. It's really radical. I'm not sure Opus could ever do anything like that by itself, and nobody else in FidoNet showed even the slightest bit of interest ... so it was an idea that just died on the vine."

-- Wynn Wagner III.

### 4.5.1 USING FTB

FTB is a set of 4-digit hex numbers that can be used instead of standard menu commands.

The format of every FTB command is: <ESC><number> or \$<number>. Both forms work.

Pressing <ESC> 1001 is Goodbye in ANY language. In user macros, the dollar sign character \$ may be used in place of the <ESC> to use FTB commands. Opus will substitute <ESC> for \$ before executing the user defined macro.

Several of the options have the same number (several \_SHOW's, for example), so, using <ESC> nnnn, how does one select among the various choices? Hmmm.

## 4.5.2 FTB COMMANDS LIST

Command.....FTB ID

\_GOODBYE ..... 1001  
 \_STATS ..... 1002  
 \_SHOW ..... 1003  
 \_YELL ..... 1004  
 \_USERLIST ..... 1005  
 \_VERSION ..... 1006  
 \_OUTSIDE ..... 1007  
 \_SYSOP\_Message ..... 1009

CHG\_AREA ..... 1008

\_MAIN\_mode ..... 10f1  
 \_CONFIG\_mode ..... 10f2  
 \_HISTORY\_mode ..... 10fa  
 \_SYSOP\_mode ..... 10f3  
 \_MESSAGE\_mode ..... 10f4  
 \_FILE\_mode ..... 10f5  
 internal\_oped ..... 10f8  
 internal\_lore ..... 10f9

\_SECTION\_mode ..... 1100

\_CUSTOM1\_mode ..... 1101  
 \_CUSTOM2\_mode ..... 1102  
 \_CUSTOM3\_mode ..... 1103  
 \_CUSTOM4\_mode ..... 1104  
 \_CUSTOM5\_mode ..... 1105  
 \_CUSTOM6\_mode ..... 1106

SET\_PWD ..... 2001  
 SET\_HELP ..... 2002  
 SET\_NULLS ..... 2003  
 SET\_WIDTH ..... 2004  
 SET\_LEN ..... 2005  
 SET\_TABS ..... 2006  
 SET\_MORE ..... 2007  
 SET\_VID ..... 2008  
 SET\_CLS ..... 2009  
 SET\_EDIT ..... 200a  
 SET\_IBM ..... 200b  
 SET\_DEF ..... 200c  
 SET\_TALK ..... 200f  
 SET\_MENU ..... 2010  
 SET\_ASKGRAPH ..... 2011  
 SET\_KEYS ..... 2012  
 SET\_GMENUS ..... 2013  
 SET\_BLOCK ..... 2014  
 SET\_TIME ..... 2015

USR\_PWD ..... 2050  
 USR\_ADDRESS ..... 2051  
 USR\_PHONE ..... 2052  
 USR\_CITY ..... 2053  
 USR\_LIST ..... 2054  
 USR\_LANG ..... 2055  
 USR\_ALIAS ..... 2056  
 USR\_BIRTHDAY ..... 2057  
 USR\_ANSWER ..... 2058  
 USR\_NAME ..... 2059

MSG\_FORWARD ..... 3001  
 MSG\_PRIOR ..... 3002  
 MSG\_PARENT ..... 3003  
 MSG\_CHILD ..... 3004  
 MSG\_NEXT ..... 3005  
 MSG\_NONSTOP ..... 3006

Command.....FTB ID

MSG\_EDIT\_NEW ..... 3007  
 MSG\_EDIT\_REPLY ..... 3008  
 MSG\_UPLD ..... 3009  
 MSG\_LAST ..... 300a  
 MSG\_LIST ..... 300b  
 MSG\_SCAN ..... 300c  
 MSG\_INQ ..... 300d  
 MSG\_KILL ..... 300e  
 MSG\_HURL ..... 300f  
 MSG\_FRWD ..... 3010  
 MSG\_BOMB ..... 3011  
 MSG\_XPRD ..... 3012  
 MSG\_QUICK ..... 3013  
 MSG\_QINQ ..... 3014  
 MSG\_RULES ..... 3015  
 MSG\_NAREA ..... 3016  
 MSG\_HELP ..... 3017

SYS\_AREA ..... 4001  
 SYS\_MAIL ..... 4002  
 SYS\_EVTNT ..... 4003  
 SYS\_PRIV ..... 4004

F\_LOCATE ..... 5001  
 F\_TITLES ..... 5002  
 F\_DSPL ..... 5003  
 F\_DNLD ..... 5004  
 F\_UPLD ..... 5005  
 F\_RAWDIR ..... 5006  
 F\_ARCDIR ..... 5007  
 F\_HURL ..... 5008  
 F\_KILL ..... 5009  
 F\_DNHANG ..... 500a  
 F\_NEW ..... 500b  
 F\_HELP ..... 500c  
 F\_TEMP ..... 500d  
 F\_LONG ..... 500e  
 F\_ADOPT ..... 500f

ED\_SAVE ..... 6001  
 ED\_ABORT ..... 6002  
 ED\_LIST ..... 6003  
 ED\_CHG ..... 6004  
 ED\_INSRT ..... 6005  
 ED\_DEL ..... 6006  
 ED\_CONT ..... 6007  
 ED\_TO ..... 6008  
 ED\_FROM ..... 6009  
 ED\_SUBJ ..... 600a  
 ED\_ATTR ..... 600b  
 ED\_DISK ..... 600c  
 ED\_UUCP ..... 600d

\_PICK\_section ..... 7000  
 \_MAKE\_section ..... 7001  
 \_MY\_section ..... 7002

\_Send\_message ..... 7050  
 \_See\_lines ..... 7060  
 \_See\_chat ..... 7061

\_MY\_Macro ..... 8000  
 \_MAKE\_Macro ..... 8001  
 \_MAKE\_Msg\_Macro ..... 8002  
 \_MAKE\_File\_Macro ..... 8003





## 5

## OPUS SUPPORT FILES

Opus CBCS requires a number of files be present for correct operation. Opus CBCS may optionally make use of a number of other files which are not required to be present. These files might be defined by the sysop or generated automatically by Opus at runtime. Many of these file names are actually definable in the Opus control file and this is indicated where appropriate in the following list. The file names listed in the comprehensive table below represent the names delivered in the default Opus installation kit. It is recommended that you do not change the names unless you have very good reason to do so.

All files created by the sysop are subject to the name and path rules for multiple language support. Multiple language support is described in the Opus Operations Manual.

Opus is very flexible in the types of files that can be used to give a customized appearance to the user, but it can be very confusing to the new sysop trying to set them up. Study the Opus login sequence in the table in section 5.1.

## LEGEND

Column 2 ("R") required file?  
 Column 3 ("C") created by?  
 Column 4 ("D") defined in BBS.CTL

Y = Yes	E = External program	Y = Yes
N = No	S = Sysop	N = No
X = Does not apply	O = Opus	X = Does not apply
! = Obsolete	X = Does not apply	! = Obsolete
	! = Obsolete	

Filename	R	C	D	Description
#####.SSN	X	O	X	These files are created by Opus when there is a CONNECT for a mail session with another system, but the mail transfer does not complete successfully for whatever reason. These flag files are incremented from .SS1 to .SS5, at .SS5 Opus will stop making calls to the relevant system until the file is deleted manually or during a special Opus house- cleaning event. The first four characters of the #####.SSN filename are the Net in hexadecimal and the last four characters are the Node number in hexadecimal.
?????#.CTL	X	E	Y	Control file created by Opus external file transfer protocol engine. The ?????? is up to 6 characters of the external protocol's program name; the last two characters are overwritten by the hexadecimal ## task number of the Opus invoking the protocol.
?????#.LOG	X	E	Y	Log file created by Opus external file transfer protocol engine. The ?????? is up to 6 characters of the external protocol's program name; the last two characters are overwritten by the hexadecimal ## task number of the Opus invoking the protocol.
ABOUT_METXT	N	S	Y	Short text file explaining brief details about your system. Sent by your system when called by another system which either requests that filename or a filename which you do not have available for request.
ACTIVE##.DAT	X	O	X	Created by Opus root directory where ##=active task number in hexadecimal. ACTIVE##.DAT is a zero length file which indicates that a user is currently online or the sysop is using keyboard mode on the Opus CBCS using the hexadecimal task number ##.
AREAINF.BBS	N	S	N	Displayed to users whenever they enter a message area. This file is placed in the particular message area directory. Useful for discouraging commercial ads in buy 'n sell echomail conferences. For multi-language systems, use AREAINF.BBS.
BAD_BNDL.yyy	X	O	X	If Opus is tossing echomail (and not some other external utility) Opus saves partially tossed packets that are grunged by renaming the grunged packet to BAD_BNDL.yyy where .yyy=.000, .001, .002, etc. so that the sysop can attempt to correct the problem and/or salvage the rest of the messages in the packet (eg by renaming the bundle to 00000000.PKT and rerunning Opus with the -u command line switch).

BADxxx.yyy	X	O	X	If Opus is extracting mail packets (and not some other external utility), Opus saves grunged mail archives by renaming the grunged archive to BADxxx.yyy where yyy=.000, .001, .002, etc. and xxx=arc, zoo, zip or lzh (ie the name of the compression method used in the archive).
BADNAME.BBS	N	S	Y	OEC file which Opus V1.70 will display to a user whose logon name matches a name or name fragment specified in the USERNAME.TXT file. If not defined in the control file, Opus will merely pretend it is a private system and log the user off.
BAD_PWD.BBS	N	S	Y	OEC file executed when a user incorrectly enters the logon password five times in a row. May be used to obtain information from the user for later verification by the sysop or may explain to user how to contact the sysop for password verification.
#####Z BADWAZOO.###	X	O	X	This file is the start of a mail file transfer that was aborted. It is used by Opus to recover from blown WaZOO mail transfers using Zmodem recovery. BADWAZOO.### is what you have so far. Information as to where the file was coming from, size and real name (etc) is stored in a file in your outbound directory with a .Z extension. The first four characters of the #####Z filename are the Net in hexadecimal and the last four characters are the node number in hexadecimal. You should have one .Z file for each BADWAZOO file. A zero length BADWAZOO means that the transfer bombed before you received a single byte. These files should not be deleted (unless BADWAZOO is zero length) or Opus will not be able to resume the file transfer on the next call where it left off the first time.
BARRICAD.BBS	N	S	Y	Displayed to users when they attempt to enter a barricaded message or file area.
BBS.CTL	Y	S	N	The master Opus control file which is E processed by NACL and created from a BBS.PRM file by CAYENNE or from scratch by the Sysop.
BBS.PRM	Y	S	N	The compiled version of the BBS.CTL file (see E above) to be read by Opus. Created from the BBS.CTL file by NACL.
BYEBYE.BBS	N	S	Y	Displayed to users when they log off.
CBBS	N	S	Y	Help file displayed to users at the Change setup menu.
CHAT##.	X	O	X	Used by Opus V1.70 to store interline chat text for line ##. Deleted by Opus when read by line ##.
COMMON.DAT	X	O	X	Used by Opus to store total callers and quote file pointer for a multi-line system.
CONFHELP.BBS	N	S	Y	asks them about how they would like to configure their setup (graphics, hitedch, ansi, ascii menus, ibm characters etc). Replaces the SHADES.BBS file in Opus V1.x.
CONTENTS.BBS	N	S	Y	Help file for archive contents display command on the menu
DAYLIMIT.BBS	N	S	Y	Displayed when a user logs on who has used up his daily time allotment.
DIR.BBS	I	I	I	Obsolete; not used by Opus V1.70. May still be used by some utilities (eg, FIDO-FAM in file areas).
DOWN.LOG	N	O	Y	File to which Opus V1.70 will optionally log all user downloads (user, date, protocol, filenames).
ECHO.CTL	I	I	I	Obsolete; not used by Opus V1.70.
ECHO.MDX	Y	X	X	Opus V1.70 index file for echomail message area echomail tag names.
ECHOTOSS.LOG	N	O	Y	File created by Opus V1.70 when a user (or the sysop) has entered echomail in a message area. The file is used by Opus for its internal echomail scanner and may be used by external utilities. The LOG ECHOMAIL option must be enabled for the file to be created by Opus.
EDITOR.BBS	N	S	Y	Help file for LORE editor shown to users the first two times that they use LORE. This file should explain automatic line wrapping and how to end entering a message (two ENTERs).
EDITHELP.BBS	N	S	Y	Help file displayed to users at the LORE editing menu.
FLBBS	- N	S	N	OEC file executed when the sysop hits the FID.BBS corresponding function key at the local console while a user _is_ online. Stored in the Opus MISC directory.
FLBAT - FLBAT	N	S	N	Batch file executed when the sysop hits the corresponding function key at the local console while a user is _not_ online. Stored in the Opus MISC directory.
FIDouser.LST	N	S	N	Sorted list of usernames and address generated by most Version 6 nodelist compilers. Opus will use this to look up an address when a username is entered at the address prompt when entering a message in the matrix/netmail area.
FILEAREA.BBS	N	S	Y	Displayed when user requests a listing of file areas.
FILEHELP.BBS	N	S	Y	Help file displayed to users at the File section menu.
FILELIST.CCC	N	S	Y	Compressed file containing a text list of files which are requestable from your Opus system. (CCC should indicate the particular flavour of file compressor which you use.)
FILENAMEDAT	I	I	I	Obsolete; in Opus V1.70 the file area "name" is specified with the AREA option in the file area definition. See further section 2 of this manual.
FILES.BBS	I	I	I	Obsolete; these listings have been replaced in Opus V1.70 by the files database.



FILESBBS.ADX	Y	E	X	The Opus V1.70 files database area index file for fast file area lookups. Created by CONFILE.EXE and maintained by OFARE.EXE and/or OPUS-FAM.
FILESBBS.DAT	Y	E	X	The Opus V1.70 files database which is created from the superseded FILES.BBS files by the CONFILE.EXE program.
FILESBBS.NDX	Y	E	X	The Opus V1.70 files database B+tree index file for fast filename lookups. Created by CONFILE.EXE and maintained by OFARE.EXE and/or OPUS-FAM.
HISTORY.BBS	N	S	Y	Help file displayed to users at the Opus V1.70 History menu.
INMAIL##.SSS	X	O	X	Created by Opus in the STATUS directory when received mail is being processed if a multi-tasking method was specified in the control file. ##=hexadecimal task number of the Opus processing mail.
INTERLIN.BBS	N	S	Y	Help file for the Opus V1.70 _SEND_MESSAGE interline chat menu option.
INQUIRE.BBS	N	S	Y	Help file for message area inquire command.
KILLDUP.DAT	X	O	X	Duplicate message information - one per echomail message area directory. Used by Opus internal echomail processing.
LASTREAD	N	O	X	Created in message directories for external message editors normally used by the sysop. Allows Opus to track lastread pointers for the sysop.
LASTUS##.DAT	X	O	X	Created by Opus in the STATUS directory when caller accesses an outside function. Contains caller's entire user record for use by external program. (##=hexadecimal task in use.)
LEAVING.BBS	N	S	Y	OEC file executed when a user leaves Opus to enter an outside DOS program.
LFILE.DAT	X	O	X	Opus V1.70 file which holds the date users last accessed a particular file area. Opus will create one of these files in each file area directory.
LOCATE.BBS	N	S	Y	Help file for file area locate command.
LOGO.BBS	N	S	Y	Displayed at connect before user logon (ASCII text only, no longer than 1K or mail sessions may fail).
LREAD.DAT	X	O	X	Opus V1.70 file which holds the last read message and highest message read pointers for users. There is a LREAD.DAT file created by Opus in each message directory.
MACRO.BBS	N	S	Y	OEC help file displayed by Opus V1.70 to users before they start to create their own user macros.
MAILBAT.BAT	X	O	X	Batch file created by Opus V1.70 when the string defined in the control file for the EXTMILER option is detected at the login name prompt. See section 2 of this manual for details of file format.
MAIN.BBS	N	S	Y	Help file displayed to users at the Main menu.
MAKE.BBS	N	S	Y	OEC help file displayed by Opus V1.70 to users before they start to make their own user defined sections.
MAYBENEW.BBS	N	S	Y	OEC file displayed to new users after the Opus "Can't find name" and before the "Register as a new user?" prompt.
MSG.BBS	N	S	Y	Help file displayed to users at the Message menu.
MSGAREA.BBS	N	S	Y	Displayed when user requests a listing of message areas.
MSGNAME.DAT	I	I	I	Obsolete; in Opus V1.70 the file area "name" is specified with the AREA option in the message area definition. See further section 2 of this manual.
MSGTMP.MSG	X	O	X	File created by Opus (sometimes by the sysop) S when the sysop is using an external editor to enter messages either locally or remotely.
MYQUEST.BBS	N	S	Y	Opus V1.70 logon question OEC file. Opus will automatically save the user's response in the user record in the Opus user file.
NAME.FDX	Y	E	X	Opus V1.70 index file for file area names, created by NACL.
NAME.MDX	Y	E	X	Opus V1.70 index file for message area names, created by NACL.
NERF.BAT	Y	S	N	The traditional name for the Opus startup batch file. Ask any grizzled Opus veteran to explain its significance to you! You may find an explanation under Opus folklore in the Opus Operations Manual if Guido ....
NEWUSER1.BBS	N	S	Y	Displayed to new users immediately after logon and before Opus prompts for a password.
NEWUSER2.BBS	N	S	Y	Displayed to new users immediately after choosing their passwords.
#####.TMP	X	O	X	File created by Opus after an aborted incoming file request (##### is digital). This file may be deleted with impunity.
NODELIST.DAT	N	E	N	Compiled Version 6 nodelist files for Fidonet NODELIST.IDX operation. Not required if your Opus does not do mail sessions.
NODEX.DAT NODEX.DAT SYSO.NDX	N	E	N	Compiled Opus V1.70 Version7 nodelist files for FidoNet operation. Not required if your Opus does not do mail sessions. (SYSOP.NDX is the Version 7 equivalent of the Version 6 FIDouser.LST).

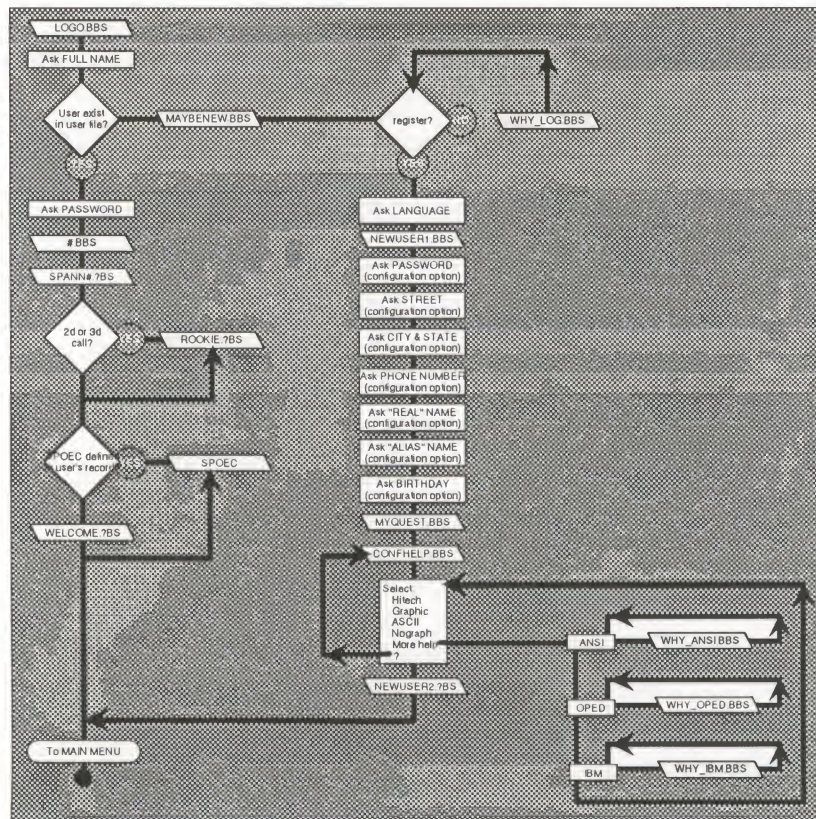
NULLDMP	X	O	X	Written out during a system crash for debugging help. Save for Opus Lower Code Dweller analysis. Do Not Open. No sysop serviceable parts inside.
NUMBER.FDX	Y	E	X	Opus V1.70 index file for file area numbers, created by NACL.
NUMBER.MDX	Y	E	X	Opus V1.70 index file for message area numbers, created by NACL.
OPUSXFER.NNN	X	O	X	A zero-length file created by Opus after a failed/aborted file transfer. May be deleted. (NNN increments for each file, starting at 001).
ORIGIN	I	I	I	Obsolete; instead Opus V1.70 uses the control file option in the message area definition (see further section 2 of this manual).
OKFILE.LST	N	S	Y	List of requestable files, or invokable procedures and programs via file request mechanism. Refer to the Matrix Section of the Opus Operations Manual for details of the format of this file.
OPED.BBS	N	S	Y	Help file for OpEd full-screen editor.
OPUS.LOG	Y	O	Y	Log file to which Opus system activity is written.
PASSWORD	N	S	Y	Contains password(s) for barricaded message or file area access (can be any name and may specify in the area definition one file for each area or a common file for all barricaded areas).
QUOTES.BBS	N	S	Y	Quotable quotations for OEC quote command (refer to section 5 of this manual for a complete listing of OECs).
REP_EDIT.BBS	N	S	Y	Displayed when the Edit option of the LORE editor is chosen; the help file explains how to add to the beginning of the line or replace text already entered on the line.
RETURN.BBS	N	S	Y	Displayed when user returns to Opus from an outside DOS program (except when EXIT method is used; see further section 3 of this manual).
ROOKIE.BBS	N	S	Y	Displayed on users' 2nd and 3rd calls.
RULES.BBS	N	S	N	Opus V1.70 displays the "rules" option on the menu in a message area if there is a RULES.BBS file in the directory to which the message area path points. For multi-language systems, use RULES.BBS. Refer to the MSG_RULES command in section 3 of this manual.
SCHED.DAT	Y	E	Y	Contains information used by Opus to schedule events and behaviour windows (created and maintained by OEVENT).
SPANIN#.BBS	N	S	N	Special announcement file (#=1-5). Number of times for it to be shown to users is set in the relevant user records in the Opus user file.
SYSFILE.DAT	Y	E	X	Opus V1.70 database file for file area definitions, created by NACL.
SYSHELP.BBS	N	S	Y	Help file displayed to the sysop at the D sysop menu.
SYSTEM.DEF	I	I	I	Obsolete.
SYSTEM##.DAT	I	I	I	Obsolete; Opus V1.70 keeps this information in the SYSMMSG.DAT and SYSFILE.DAT database files.
SYSMMSG.DAT	Y	E	X	Opus V1.70 database file for message area definitions, created by NACL.
TIMEWARN.BBS	N	S	Y	Displayed to user after logon when there is an impending event which will cut short the time they would otherwise be allowed for the session.
TOOSLOW.BBS	N	S	Y	Displayed to caller whose baud rate is below minimum required for logon as set in configuration.
UP.LOG	N	O	Y	File to which Opus V1.70 will optionally log all file uploads (user, date, protocol, filenames).
USER.DAT	Y	O	Y	File containing the Opus user file (names, passwords, statistics, etc). Created by the Opus V1.70 installation utility and maintained with OUFM.EXE.
USER.IDX	I	I	I	Obsolete.
USER.NDX	Y	O	Y	Opus V1.70 B+tree index into the Opus user file for fast user record name/alias lookups. Initially created by USERNDX.EXE and maintained by Opus and OUFM.EXE.
USERNAME.TXT	N	S	Y	Contains unacceptable new user names or name fragments. See section 2 of this manual for details of file format.
UUCPLIST.TXT	N	S	Y	Text file containing UUCP names and network addresses. See section 2 of this manual for details of file format.
WELCOME.BBS	N	S	Y	Shown immediately after logon.
WHY_ANSI.BBS	Y	S	N	Help file for new user ANSI graphics question.
WHY_FB.BBS	Y	S	N	Help file for logoff comment (feedback) to sysop question.
WHY_HU.BBS	Y	S	N	Help file for logoff (hang-up) confirmation question.
WHY_IBM.BBS	Y	S	N	Help file for new user IBM-PC characters question.
WHY_LOG.BBS	Y	S	N	Opus V1.70 help file for logon "Register as a new user?" question.
WHY_OPED.BBS	Y	S	N	Help file for new user OPED editor question.
WHY_PVT.BBS	Y	S	N	Help file for private message entry question.
WHY_SHAD.BBS	I	I	I	Obsolete; for Opus V1.70, refer to the CONFHELP.BBS file instead.
#.BBS	N	S	N	Custom welcome file for user whose record number is # (displayed to user at logon).



XDATGONE.BBS	N	S	Y	Displayed when user's expiry date has arrived or passed.
XDATWARN.BBS	N	S	Y	Displayed when user's expiry date is within the number of days specified in the control file by the EXPIRE DAYS option. See further section 2 of this manual.
XFERBAUD.BBS	N	S	Y	Displayed to caller whose baud rate is below the minimum required for file download as set in the control file.
XTIMGONE.BBS	N	S	Y	OEC file displayed to users when their expiry time has arrived.
XTIMWARN.BBS	N	S	Y	OEC file displayed to users when their expiry time is within the number of minutes specified by the EXPIRE MINUTES control file option. See further section 2 of this manual.
xxxxxxx.MNU	Y	E	Y	Language file for menus (eg ENGLISH.MNU) (user configurable), created by NACL.
xxxxxxx.PRM	Y	E	Y	Control file compiled by NACL (eg BBS.PRM), created by NACL.
xxxxxxx.SYL	Y	X	Y	Language files for system prompts (eg ENGLISH.SYL, ENGLISH.USL), created by NACL.
xxxxxxx.USL	Y	X	Y	Language files for user prompts (eg ENGLISH.SYL, ENGLISH.USL), created by NACL.
YELL.BBS	N	S	Y	OEC file displayed to users who yell for the sysop when yell is turned off in the schedule file or there is no answer when yell is enabled.

## 5.1 OPUS LOGIN SEQUENCE

The sequence of events, and files shown, when a new user is at login is depicted in the following figure.



Opus Login Sequence



## 6

## COMMAND LINE SWITCHES

Opus has several command line switches that can be passed to it for various purposes. They may be passed in any order.

Switch	Meaning	Comment
-m"string"	External Mailer load. Example: -m'btex share'	This overrides the load mailer command, current mailer event and control file settings
-b#####	Pass the speed of the communications port to Opus, where ##### is the speed in characters per second. Example: -b9600 means 9600 bps.	Used to pass control from a front end mailer such as Binkleyterm, FrontDoor, or D'Bridge. This parameter also implies that there is already a live connection (carrier detect present).
-d	debug - shows memory This will use up disk space utilization as program in a HURRY! You should only runs. use this option if you are helping debug a memory allocation problem in Opus.	
-e	Do not run forced events This switch is useful if you that are past due. are doing some testing at the keyboard and do not want forced events to cause an exit while you are testing.	
-k	Keyboard mode	Overrides setting in control file. Does not initialise a communications port.
-nL	Do not show user list when creating a message.	Same as "Edit Disable UserList" in control file. Overrides control file setting.
-nO	No OpEd Editor	Guido will get you!
-o	Quick relog from outside - bypasses welcome and introduction screens.	Used for return from an Opus external exit at a menu.
-p#	Specifies the com port to use.	Normally passed by frontend mailer. This overrides the OUTPUT setting in the Opus control file.
-r###	Set refund percentage for uploads	Same as "Refund <number>" in control file. Overrides control file.
-s	Force Opus V1.70 to do an echo scan right now.	If echo messages have been entered by an external editor, this will get Opus to pick them up.
-t###	Pass the time in minutes before the next event is scheduled in the front end mailer. Example: -t25 means 25 minutes remain until the next event is scheduled.	Opus will allow the user to remain online until the end of his allocated time, or until the next scheduled event from the mailer, whichever represents the SHORTER time
-u	Unpack any mail bundles found now.	Do not place in your batch file as the only way of invoking Opus, or it will keep trying to unpack mail forever!

This stuff is so technical,  
I wouldn't let my mother even see the cover page.

-- Bob Davis  
Frustrated Opus 1.1x Doc Writer

# MEMORANDUM FOR THE RECORD

DATE: 10/10/1964

TO: SAC, NEW YORK

FROM: SAC, NEW YORK

SUBJECT: [REDACTED]

RE: [REDACTED]

1. [REDACTED]

2. [REDACTED]

3. [REDACTED]

4. [REDACTED]

5. [REDACTED]

6. [REDACTED]

7. [REDACTED]

8. [REDACTED]

9. [REDACTED]

10. [REDACTED]

11. [REDACTED]

12. [REDACTED]

13. [REDACTED]

14. [REDACTED]

15. [REDACTED]

16. [REDACTED]

17. [REDACTED]

18. [REDACTED]

19. [REDACTED]

20. [REDACTED]

21. [REDACTED]

22. [REDACTED]

23. [REDACTED]

24. [REDACTED]

25. [REDACTED]

26. [REDACTED]

27. [REDACTED]

28. [REDACTED]

29. [REDACTED]

30. [REDACTED]

31. [REDACTED]

32. [REDACTED]

33. [REDACTED]

34. [REDACTED]

35. [REDACTED]

36. [REDACTED]

37. [REDACTED]

38. [REDACTED]

39. [REDACTED]

40. [REDACTED]

41. [REDACTED]

42. [REDACTED]

43. [REDACTED]

44. [REDACTED]

45. [REDACTED]

46. [REDACTED]

47. [REDACTED]

48. [REDACTED]

49. [REDACTED]

50. [REDACTED]



**7****EXTERNAL PROGRAM USAGE**

BBS, *n.* A loose collection of usually unrelated, sometimes uncooperative or caustic, communications elements, flying in close formation.

--- from Wynn's Glossary Of BBS Terms

Opus CBCS allows the user to invoke external programs using various methods. These external programs fall into several categories. They can be used to add variety and character to your CBCS, provide remote access for the sysop to external maintenance functions, and add new file transfer protocols in the file section. Opus provides certain mechanisms to exchange data between itself and the external programs. This section is devoted to explaining the methods and rules for program invocation, data interchange, and return to Opus after the program runs.

**7.1 INVOCATION FROM MENUS**

The Opus menus can have commands added to them for invoking external programs. If you build your menus with NACL, there are five menu commands that can be used to invoke external programs.

```
_OUTSIDE  <priv>  !<locks>  "Command"    = SRUN <Program>
_OUTSIDE  <priv>  !<locks>  "Command"    = RUN  <Program>
_OUTSIDE  <priv>  !<locks>  "Command"    = SDOS <Command>
_OUTSIDE  <priv>  !<locks>  "Command"    = DOS  <Command>
_OUTSIDE  <priv>  !<locks>  "Command"    = EXIT <errorlevel>
```

**7.1.1 SRUN/RUN COMMAND OPTION**

When you use the RUN command, Opus will pass a series of parameters to your external program. The external program must be a .EXE or .COM file and a full path specification must be used. The parameters passed will change depending on whether the program is called from the sysop's keyboard or from a remote operation.

If local keyboard mode:

```
program.ext [other parms in menu] -k -t# -m# -f# -r#
```

If remote:

```
program.ext [other parms in menu] -p# -b# -t# -m# -f# -r#
```

where,

Parm	Description
-k	Keyboard mode
-p#	Active Port number (0=Com1, 1=Com2, etc.)
-b#	Active bps connect speed (i.e. -b2400)
-t#	Active Task Number
-m#	Active message area number
-f#	Active file area number

-r#            Number of minutes remaining for this user

All numbers are unsigned integers, not hexadecimal!

The "[other parms in menu]" are the commands you set with NACL or whatever. Do NOT use periods in external program command lines! For example, "/N:lastus01" is OK, but "/N:lastus01.dat" corrupts memory.

The external program will probably see the program name and parameters, ie [other parms in menu], as one big argument. It will be up to the external program to parse that 0'th parameter.

For example, suppose we call into our system with these settings:

```
port           =      0
connect speed  =     2400
task no.       =      2
last msg area  =      4
last file area =      8
time remaining =     85
```

The MAIN menu has the option configured as:

```
_OUTSIDE Clerk !ABCD "Manager" RUN C:\OPUS\PROGRAM.EXE YEP
```

This example would present the command "Manager" in the menu to anyone with Clerk privilege and User Keys ABCD. If selected, Opus would attempt to run C:\OPUS\PROGRAM.EXE and pass the command line:

```
"C:\OPUS\PROGRAM.EXE YEP -p0 -b2400 -t3 -m4 -f8 -r85"
```

### 7.1.2 SDOS/DOS COMMAND OPTION

When you use the DOS option, Opus calls a copy of COMMAND.COM and passes the command line exactly following the command syntax for the RUN OPTION. You only need to use DOS when you want to run a batch file or make use of COMMAND.COM's human features. Some programs may require that you use the DOS option. If in doubt, and the program's author gives no hints, try the DOS option.

What is the difference between "SDOS" and "DOS"? Normally Opus will try to reload LASTUS##.DAT after an external program if the SECURE level has been set greater than 0. Sometimes you do need to re-read LASTUS##.DAT, sometimes you do not. SDOS tells Opus V1.70 to do the external program, but do NOT re-read LASTUS##.DAT afterwards. It will make returning to Opus a little faster but Opus will not know about any changes to LASTUS##.DAT. This is also useful if you suspect some external program may be changing things that should not be changed.

### 7.1.3 EXIT COMMAND OPTION

With the EXIT command option, Opus exits completely with a DOS errorlevel specified by the menu command definition. This option is used for cases where the sysop wants to have as much memory available as possible. Some information for use by external programs is available from the LASTUS##.DAT file. The EXIT option provides virtually no coupling between Opus and any external programs or procedures. It is the sysop's responsibility to ensure that the security of the system is not compromised when this option is used.

Opus has an option in the control file called "REBOOT". You should enable this option or use an external watchdog program to monitor carrier from your modem after Opus has performed the exit function. If carrier is lost while outside, the system will reboot. Otherwise, another user could call in and gain access to whatever external program was active at the time of the loss of carrier. Of course, this could get a little messy if you are running a multi-tasker and three lines. The reboot would bring down the other two systems, too. If you are doing a three line system, you probably know what you are doing anyway.

Returning to Opus after an external exit without hanging up the modem requires a special set of command line parameters. To return to Opus, use this command:

```
OPUS <control file name> -o
```

Where,

- o specifies that an abbreviated logon sequence will occur going straight to the main menu.

Depending on the setting of the RELOG option in the control file, Opus may (0) hang up; (1) ask user for name and password and check it against the LASTUS##.DAT file and USER.DAT file; (2) ask user for password only and check it against LASTUS##.DAT file and USER.DAT file; or (3) not ask any questions, just return user to exactly where they were before they exited Opus. NOTE: you must also set the AFTER RELOG EXIT option in the control file for relogging to work correctly.

---

## 7.2 INVOCATION FROM EMBEDDED COMMAND FILES

You may invoke an external program from any file shown by Opus using one of the Opus embedded commands to run a program. This invocation is equivalent to the DOS/SDOS command option from the menus, except that the only information passed is that specified in the embedded command file. The port, speed, message area, file area, task number, and time remaining data is not passed. The command line specified in the embedded command is passed literally with no interpretation of language or task number as is used elsewhere in embedded commands. You can use the embedded command method to run programs with Opus that may otherwise not run from an Opus menu.

An example of an embedded command would be:

```
^OCDOORWAY COM1 /S:* /G:ON /K:600 /V:D^U /M:1000 /B:Z /O:T /C:DOS
```

This command would run DOORWAY.EXE and pass the command:

```
"COM1 /S:* /G:ON /K:600 /V:D^U /M:1000 /B:Z /O:T /C:DOS"
```

---

## 7.3 RELOADING LASTUS##.DAT

When Opus calls an external program by the DOS or RUN (not SDOS nor SRUN) method, or using an embedded code which rereads LASTUS##.DAT, it will optionally read in the information posted in LASTUS##.DAT for checks and update the user record in memory to write back to the user record. This allows an external program to update the User record online. How much information is read and checked is determined by the option in the control file called "SECURE <number>". These options are:

- 0 Never re-read LASTUS##.DAT, ignore it.
- 1 Just copy the information from OPUS\_id on, no Opus information is copied.
- 2 Check the Name, Privs, Keys, and OPUS\_id before copying the complete user record.
- 3 Just check the OPUS\_id and live dangerously.

Here are some answers to questions that might arise about these security options:

- Q: I would like for Opus to always re-read the LASTUS##.DAT data. How am I "living dangerously" by specifying SECURE 3?
- A: Someone could write a trojan program that would assign a particular or all users SYSOP privileges and turn on all their KEYS.
- Q: What is an OPUS\_id?
- A: Its the string "OPUS" that starts at byte 0x300 in the user record. Its just there to make sure that the file has not been munged too much.



- Q: What is the "Opus information" mentioned for number 1?
- A: "Opus Information" would be all the non-external stuff, from the user name through the `sy-sop_comment`.
- Q: What happens if the "check" in number 2 fails?
- A: It is logged as an error and the user record in memory, and therefore the one in `USER.DAT`, is not updated by Opus.

---

## 7.4 EXTERNAL FILE TRANSFER PROGRAMS

Opus CBCS has a suite of commonly used file transfer protocols built-in. It is recognized, however, that file transfer protocols evolve and change. As such, provision is made to add up to sixteen additional file transfer protocols for use in the Files Section of your CBCS.

### 7.4.1 INSTALLATION OF EXTERNAL PROTOCOLS

Installation of external file transfer protocols is very easy, provided that the author of the transfer module has followed Opus external protocol conventions described in section 7.4.2. The only precaution you need to follow that the first letter of the protocol program is a letter (A-Z) and does not conflict with other internal or external protocols. The Opus built-in file transfer protocol menu is shown here.

```
Z) modem
X) modem
Y) xmodem/1k
M) odem7
S) ealink
Q) QUIT
```

Note that the letters Z,X,Y,M,S,T (for Telink, not shown), and Q are used. That means that you must choose transfer protocol names that begin with other letters. Let us say you wish to add a Kermit protocol to your system. The transfer program is `C:\OPUS\KERMIT.EXE`. In order to add it, you would add the statement

```
EXTERNAL PROTOCOL C:\OPUS\KERMIT.EXE
```

to your Opus control file in the SESSION SECTION. After you recompile your control file with NACL, the protocol is automatically included. Your new file transfer protocol menu presented to the user would look like this.

```
Z) modem
X) modem
Y) xmodem/1k
M) odem7
S) ealink
K) ermit
Q) QUIT
```

### 7.4.2 EXTERNAL PROTOCOL LINKAGES WITH OPUS

When an external protocol is called by Opus, several pieces of information must be passed between Opus and the protocol program. These information exchange mechanisms are described in sections 7.4.2.1 through 7.4.2.3.

#### 7.4.2.1 PROTOCOL COMMAND LINE PARAMETERS

Information is passed from Opus just like calls to other programs, except the fully qualified program name is passed as the final parameter without an extension.



Program.ext [other parms] -p# -b# -t# -m# -f# -r# Program

where,

-p# Active Port number (0=Com1, 1=Com2, etc.)  
 -b# Active bps connect speed (i.e. -b2400)  
 -t# Active Task Number  
 -m# Active message area number  
 -f# Active file area number  
 -r# Number of minutes remaining for this user

All numbers are decimal. For example, Opus might call Kermit with this command:

C:\OPUS\KERMIT.EXE -p1 -b4800 -t1 -m3 -f16 -r86 C:\OPUS\KERMIT

which would mean execute KERMIT.EXE for

Port 1 (ie COM2:)  
 4800 bps  
 Task 1  
 Message area 3  
 File area 16  
 86 minutes remaining  
 C:\OPUS\KERMIT is path\filename for control and log files

#### 7.4.2.2 PROTOCOL CONTROL FILE

Opus creates a control file with information in it just prior to executing the protocol program. The control file will be located in the same directory as the protocol program. The naming convention for this control file is ProgramName.CTL, where ProgramName = Protocol

Na##, where the last two digits of the filename are replaced with the hexadecimal task number. Here are some examples.

Protocol Program	Task	Control File
C:\OPUS\KERMIT.EXE	0	C:\OPUS\KERM00.CTL
C:\PROTOCOL\KERMIT.EXE	3	C:\PROTOCOL\KERM03.CTL

The contents and format of this control file are:

Port <a>  
 Baud <b>  
 Modem <c> <d> <e> <f> <g>  
 Time <h>  
 Log OpusLogName  
 Messages MessagePath  
 Uploads UploadPath  
 Downloads DownloadPath  
 Files ListPath (only if declared in Opus control file)  
 Get|Send Transferred\_Filename (This line repeated  
   for each file to transfer)

where,

<a> = comport (1=com1, 2=com2)  
 <b> = caller baud  
 <c> = comport (1=com1, 2=com2)  
 <d> = If STEADY max\_baud else current baud  
 <e> = handshake mask  
 <f> = carrier mask  
 <g> = caller baud

<h> = time remaining in minutes  
 Get|Send = 'Get' if file is an upload to Opus  
           'Send' if file is a download from Opus

NOTE: Parameters <c>, <d>, <e>, <f>, and <g> are hexadecimal numbers. All other numbers are decimal.

An example control file for our KERMIT program might look like

C:\OPUS\KERM01.CTL for upload:

```

Port 2
Baud 2400
Modem 2 960 1 80 960
Time 320
Log R:\OPUS.LOG
Messages M:\MEMBERS\
Uploads F:\UPLOAD\
Downloads F:\OPUS110\
Get F:\UPLOAD\OEXE_111.ZIP      (This line repeated for
                                each file to transfer)

```

C:\OPUS\KERM01.CTL for download:

```

Port 2
Baud 2400
Modem 2 960 1 80 960
Time 335
Log R:\OPUS.LOG
Messages M:\MEMBERS\
Uploads F:\UPLOAD\
Downloads f:\opus110\
Send f:\opus110\OEXE_111.ZIP   (This line repeated for
                                each file to transfer)

```

#### 7.4.2.3 PROTOCOL LOG FILE

After the file transfer is complete, the protocol program must create a log file. This log file is used by Opus to write information to its own log. The logs are placed in the same directory as the control file with the same naming convention, i.e., if there is a C:\OPUS\KERM01.CTL file, then there should be a corresponding C:\OPUS\KERM01.LOG file. The log file only has one line in it for each file transferred. It is of the form:

```
Got|Sent path\filename #####
```

where,

```

Got|Sent =   'Got' for an upload to Opus
             'Sent' for a download from Opus

#####      =   Size of file in bytes

```

For Upload, the log file might look like:

```
Got f:\upload\OEXE_111.ZIP 138688
```

For Download, the log file might look like:

```
Sent f:\upload\OEXE_111.ZIP 138688
```

**8****EXITS AND ERRORLEVELS**

DOS errorlevels are used by Opus to communicate with DOS batch files when Opus exits from memory. The reason for exit may be caused by scheduled events, menu selections, external mailer exit, or normal exit after a caller, file upload, mail processing, or matrix session.

**8.1 PREDEFINED ERRORLEVELS**

There exist several fixed DOS errorlevels that will occur and override any other exit errorlevel. These are:

Errorlevel	Meaning	Action in batch file
0	Should Never Happen	RECYCLE
1	Keyboard exit (^C or Alt-Q) FILES missing during file request	HALT
2	General Errors, such as modem not reporting "OK".	RECYCLE
3	Very Serious Problem such as: * missing menu file; * failure to open or read user file or schedule file; * fossil not installed; * no response from modem after 5 tries to initialise by Opus; * ANY problem bringing the system up (main init routine).	HALT
4	Reserved	RECYCLE
5+	Sysop defined	-----
255	If no external mailer is loaded, then this is an internal MicroSoft library error.  If an external mailer is loaded, then this is trapped from the mailer exit and is assumed to mean a mailer failure, in which case Opus will not reload the mailer.	RECYCLE

Refer to the Matrix section of the Opus Operations Manual for details of building batch files to trap these errorlevels and taking appropriate action.

---

## 8.2 EVENTS

External Events will cause Opus to exit with a specified errorlevel at a particular time and date. The DOS errorlevel is specified in the specification for each type 'X' event. This errorlevel may be modified by the errorlevel offset (ie, the offset is added to the external errorlevel exit) during the current matrix behaviour window (Z-event). Refer to the section of the Opus Operations Manual on the Opus Event System for full details.

---

## 8.3 EXTERNAL MAILER

If an external mailer is loaded under Opus (refer to the MAILER COMMAND, MAILER LOAD and MAILER EXIT options in the Opus control file which are described in section 2 of this manual), it will exit with various errorlevels. Opus traps these errorlevels and takes appropriate action. If the mailer exits with an errorlevel other than a valid baud rate / 100, one of two things will happen:

- If the errorlevel = 255, Opus will assume the mailer quit (perhaps because you want a human-only window). Opus will no longer load the external mailer but run itself as usual.
- In any other case, Opus will exit with the same errorlevel, plus the behaviour window offset, as described in section 8.2.

Suppose BinkleyTerm is loaded under Opus as an external mailer and is configured to exit with errorlevel 30 on receipt of compressed mail. When Binkley exits with errorlevel 30, Opus traps the errorlevel and exits with the same errorlevel 30, passing control to the batch file to process the compressed mail. For further details of the operation of external mailers with Opus, refer to the section of the Opus Operations Manual dealing with External Mailers for further details.

---

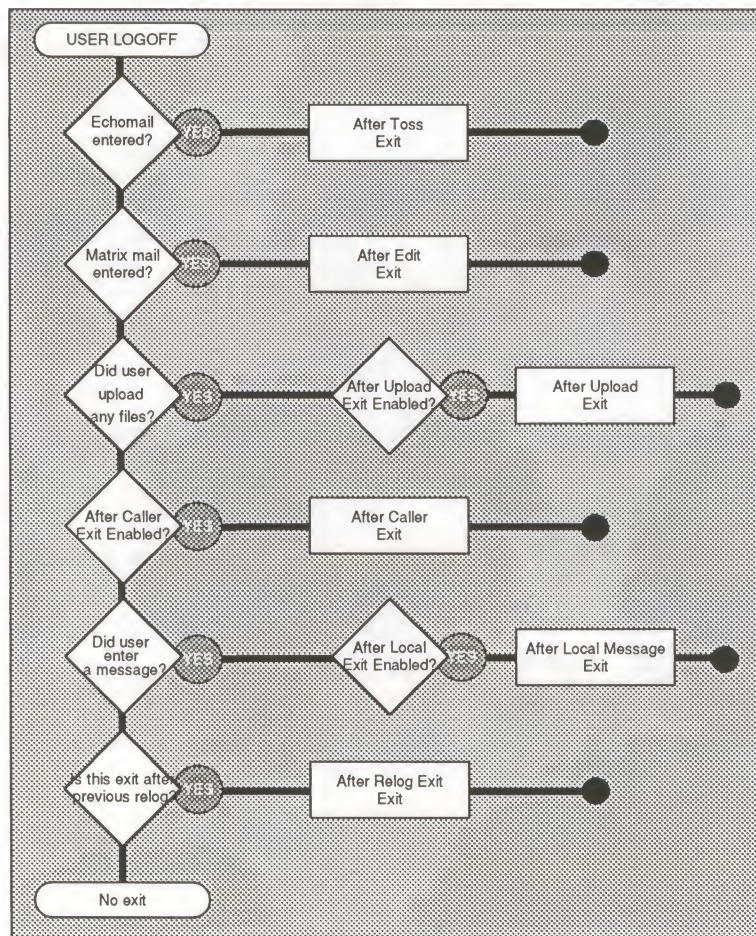
## 8.4 MENU SELECTIONS

It is feasible to configure Opus menu commands to execute an exit to DOS. The appropriate exit errorlevel is specified for each individual menu command. Refer to section 3 of this manual for full details of the various \_OUTSIDE menu options available.



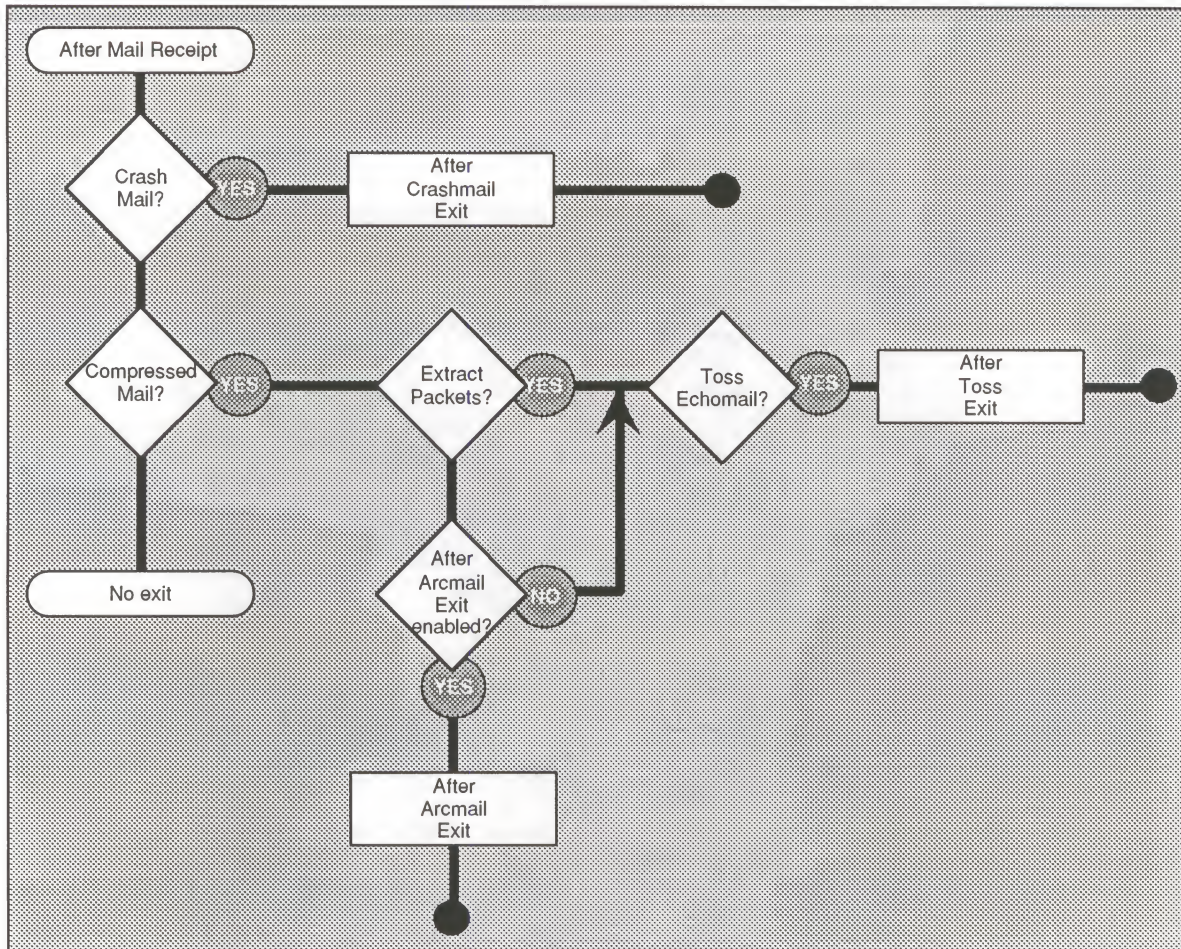
## 8.5 PRECEDENCE OF EXITS

The behaviour of Opus on exit can be quite confusing because there are so many paths that can be taken. Basically there are only two cases to consider. These are (1) after a user logs off and (2) after receipt of mail or files from a mail session. These are illustrated by the flow charts in the following figures.



Opus Exit Procedure After A Caller





Opus Exit Precedence After Mail Receipt

## 9

## ERROR MESSAGES

Error messages from Opus are often cryptic and use very compact abbreviations. This is because Opus runs in a small memory model and memory to hold message text is precious.

Error messages will indicate five major causes of problems. They are errors in setup, hardware errors, memory shortage errors, and communication errors. When an error occurs it very important to report the exact text that Opus produced. The programmer needs to know, in order to trace the problem. Many errors messages are logged in the Opus log file. If you have persistent errors, set your log to the "VERBOSE TRACE" mode and check it. Log file segments are very helpful to the programmers in finding problems.

ERROR MESSAGE TEXT	MEANING	SEVERITY
!ERR MAIN	Could not allocate enough memory to even get started.	Fatal
!ERR:FORK	Could not allocate enough memory to make command line for external program.	Fatal
!ERRcdta	Could not allocate memory for common data file.	Fatal
!FuIxOpEr:IxOFF	Could not open the user index file.	Warning
!FuIxRdEr:IxOFF	Error reading the User record.	Warning
!FuIxSkEr:IxOFF	Could not Seek to the right position in the user index file.	Warning
!FuIxSzEr:IxOFF	The sizes of the user index file and user file are not in sync.	Warning
!FuNoIxBufMem:IxOFF	Could not allocate memory to read 256 keys at a time.	Warning
!Grunged Header	In attempting to toss mail, Opus encountered an improperly formatted or corrupted header in a message bundle.	Warning
!No Fossil	A level 5 fossil driver is not loaded such as XDD, BNU, or Opuscomm.	Fatal
!OEC ERR: unknown ^p	Opus tried to display a text file that had an unknown embedded command starting with control-P. Check the file for typographical errors.	Warning
!OEC ERR: unknown ^x	Opus tried to display a text file that had an unknown embedded command starting with control-X. Check the file for typographical errors.	Warning
!PuIxOpEr:IxOFF	Could not open user index file. Gone?	Warning
!PuIxSkEr:IxOFF	User index file seek error.	Warning
!PuIxSzEr:IxOFF	Size of user index file does not match user file.	Warning
!PuIxWrEr:IxOFF	User file index write error.	Warning
!PuLkEr:LkOFF	User file record locking did not work	Warning
!PuULkEr:LkOFF	User file unlock error.	Warning
#MEMORY Heap	Not enough free memory left to go on.	Fatal
*Finished partial file "filename"	A matrix connection was re-established and the remainder or a partial file from a previous connection has been completed.	Information
*Partial file "filename"	A matrix transfer was aborted and Opus has saved a partial file, plus information to resume, should the connection be re-established.	Warning
:Mailer exit code:	Using external mailer, mailer returned unrecognized connect rate or errorlevel.	Fatal
^O:cls	Could not find next file when doing a OS<nextfile>	Warning
Cannot find overlay	DOS 2.x - default directory was changed to something other than Opus root directory prior to invoking Opus or upon return to Opus from external program or batch file.	Fatal
	DOS 3.x, 4.x - symptom of something seriously wrong with DOS's memory. Could be out of memory. Could be an undetectable environment overflow. Could be that the hard disk is too fragmented.	
ChDir	Error going to path listed for Echo Scan.	Warning
Close "string"	Self explanatory concerning any of the many Opus support Read "string" files, such as, "Could not open WELCOME.BBS missing"	Warning
Could not Open User file	Disk error	Fatal
DISK SPACE ALERT: Netfile	Disk space in your Matrix inbound file area is less than 300kbytes.	Warning
DISK SPACE ALERT: NetHold	Disk space in your outbound area is less than 300kbytes.	Warning
DISK SPACE ALERT: NetMsgs	Disk space in your Matrix Message area is less than 300kbytes.	Warning



ERR nidx: Not enough core	Insufficient memory exists for the nodelist index. Take action to make more memory. Typically, the partition size is too small in a multi-tasker when this happens.	Fatal
ERR:"string"	A generic no-memory error message. The 'string' will indicate the program module in Opus that generated the error.	Fatal
ERR:RLAST	Could not reload LASTUS##.DAT file because of memory problem.	Fatal
Exists "string"	Self explanatory concerning any of the many Opus support Read 'string' files, such as, 'Could not open WELCOME.BBS missing'	Warning
Find "string"	Self explanatory concerning any of the many Opus support Read 'string' files, such as, 'Could not open WELCOME.BBS missing'	Warning
Fprint	Could not save the message just entered.	Warning
Increase	Could not increase caller count while trying to access non-existent Common Data file.	Warning
Load	Could not load external mailer.	Warning
LOGERR	Could not re-open OPUS.LOG after external program ran.	Fatal
LOGERR	Could not re-open the LOG file after a spawn or system.	Fatal
Mark RECD	Could not mark a message as 'Received'.	Warning
MdmErr:"string"	Indicates some kind of error reported by the fossil driver. Typical strings are PARITY, OVERRUN and FRAMING. Opus faithfully reports whatever the fossil tells it.	Warning
MEM:"string"	see ERR 'string'	Fatal
MEM:lang	Could not allocate enough memory to load one of the language files.	Fatal
MEM:nupk	Could not allocate memory for tossing echo mail.	Fatal
MEM:ulbuf	Could not allocate buffer for SEALink receive.	Fatal
MENU ERROR	Could not read in the menu requested. Missing?	Fatal
No space left on device	In an attempt to write something to disk, Opus detected that the disk was full.	Warning
Not enough core	Insufficient memory exists or memory is too corrupted to run. This error will also be generated if external programs are specified in certain ways. Do NOT use periods in external program command lines! 'fsechat.exe' is OK. 'N:lastusOI' is OK. but 'N:lastusOI.dat' corrupts memory.	Fatal
Not TYPE 2 msg	While tossing mail, Opus detected that a message did not conform to Fidonet Technical Standards Committee Type 2 Message structure.	Warning
Open "string"	Self explanatory concerning any of the many Opus support Read 'string' files, such as, 'Could not open WELCOME.BBS missing'	Warning
Out of disk space on target	In an attempt to write something to disk, Opus detected that the disk was full.	Warning
Out of disk space!!	In an attempt to write something to disk, Opus detected that the disk was full.	Warning
Process	Could not make lastus##.dat.	Warning
Received:	Opus received no files or mail in a matrix transaction.	Warning
dissleysquat		
Run	External program returned error if > 0.	Warning
Seek "string"	Self explanatory concerning any of the many Opus support Read 'string' files, such as, 'Could not open WELCOME.BBS missing'	Warning
Tag	Could not write the end of the message.	Warning
TOO MANY ERRORS	Message during file transfer or matrix transaction. This indicates there is too much noise on the phone line, or the other computer is not cooperating in the file transfer.	Warning
Unexpected EOF in MsgHdr	While tossing mail, Opus detected an End of File, before the end of the last message was encountered. This is likely due to a corrupted mail bundle received from another system.	Warning
Unlink	Could not remove the ACTIVE## file before exit.	Warning
Unlisted system	A matrix address is encountered that is not in the nodelist.	Information
Update	Could not move Quote pointer.	Warning
Use	Error opening Common Data file or Event header.	Warning
Write "string"	Self explanatory concerning any of the many Opus support Read 'string' files, such as, 'Could not open WELCOME.BBS missing'	Warning



# 10 MATRIX SCRIPT KEYWORDS

Here are the keywords used with Matrix Scripts. See the Opus Operations Manual for a description of how to use scripts.

<b>AREACODE</b>	Transmits all data in the compiled nodelist phone number entry between the closing quote of the script name and the first dash in the phone number. This string can be any length.  Example:   areacode
<b>BAUD</b>	Sets the baud rate (bps) for the call to the value specified. The argument specifying the baud rate to be set is optional. If the argument is omitted, the baud rate is set to the speed given for the target node in the compiled nodelist. If you use the BAUD statement in a matrix dialing script, it needs to come after any INIT statements, since INIT will reset the baud rate to the maximum your system can handle (as set in your Opus configuration file).  Example:    baud 2400 baud
<b>BPSxxxx</b>	Allows branching or actions based on current bps rate.  Example:    If bps2400 HighSpeed
<b>CARRIER</b>	If there is no carrier when Opus processes this keyword, the script will abort.  Example:    carrier
<b>DIAL</b>	Transmit whatever information follows on the same line of the script, then wait for a modem response. If the modem reports any kind of failure (e.g., "BUSY"), the script will be aborted. NOTE: The dial prefix and suffix from the BBS.CTL file are NOT used here. You must supply your own. If no phone number is specified, the local phone number portion of the compiled nodelist entry ("xx.x"xxx-@@@- @@@@) will be used. The modem is not reset by Opus if you supply a phone number.  Example:    dial 555-1212
<b>DOS</b>	Send a command to DOS. You can process something, or run another program.  Example:    DOS DIR DOS PKZIP -m test *.pkt
<b>GOTO</b>	Causes the script processor to jump to the location in the script pointed to by a <label>. If the <label> does not exist, the script aborts.  Example:    goto try1200 <- would jump to ":try1200"
<b>IF</b>	If a match for <pattern number> was received during the last "wait" interval, transfer control to the point in the script identified by <label>. If a match is not received, control continues to the next statement in the script. Can be used any time prior to the next "wait" statement.  Example:    if 0 tryagain if bps1200 MedSpeed
<b>INIT</b>	Go through the normal Opus modem initialization routine. Resets the baud rate to the maximum baud rate your system can handle (as set in your Opus configuration file).  Example:    init

<b>PATTERN</b>	<p>Designate a text string to be searched for by the WAIT command. Up to 8 such text strings, numbered 0-7, may be searched for simultaneously. Each string must be a single word (NO embedded spaces) up to 20 characters long. Matching IS case-sensitive -- a pattern will only be matched by an IDENTICAL incoming string.</p> <p>Example:    pattern 0 :              pattern 1 OPUS              pattern 2 (clears/disables pattern 2)</p>
<b>PHONE</b>	<p>Transmit all data in the compiled nodelist phone number entry after the first dash in the phone number following the closing quote of the script name. This string can be any length.</p> <p>Example:    phone</p>
<b>SESSION</b>	<p>In most cases, this will be the last keyword in your script. It tells Opus to initiate a network session with the remote system. Opus will move through the SYNC procedure into the exchange of packets and files.</p> <p>Example:    session</p>
<b>SPEED</b>	<p>Sends the current bps rate at which the script is operating, in hundreds of bps ("12", "24", etc.)</p> <p>Example:    speed</p>
<b>WAIT</b>	<p>Wait for any of the text strings previously designated by PATTERN to be received from the remote system. This command will stay in effect until either a match is found, or there is no input from the remote system for the specified number of seconds. (The default is 40 seconds if no time is specified.) If a match is not found, the script aborts.</p> <p>Example:    wait              wait 20</p>
<b>XMIT</b>	<p>Send a command to the modem. You may also use the following translate characters:</p> <p>     ~ slight pause between commands        transmit a carriage return &lt;cr&gt;</p> <p>Example:    xmit ATZ               xmit AT ~ATH0 </p>

**11****LOG STRUCTURE**

```

+-----+
| Because there is no standardized log structure among |
| bulletin boards, we've decided to make one. OPUS will |
| respect this format. If anybody wants to join us, feel |
| free. |
+-----+

```

```

-- Wynn Wagner III, 1986
+-----+

```

**11.1 THE LOG FILE**

Programs are not required to share a log with any other process that may be running. That means that from the time a program begins until the time it exits (or forks or execs to another program), it is the sole owner of its log file.

NOTE: If you are running two or more copies of Opus with something like DoubleDOS or DESQview, it is very important that you use separate log files for each copy of Opus.

Programs should flush and close the log file before exiting or forking to another program or before bringing up a copy of COMMAND.COM. In other words, it *\*IS\** legal for a child process to use the same log file.

NOTE: Some programs (eg. SEAdog) apparently do not flush or close their log before bringing up a second or third copy of COMMAND.COM. It is unwise to have Opus share the same log with SEAdog.

**11.2 LOG LINE ENTRIES**

Each log line entry consists of these parts:

```

SYMBOL..... ! important entry, usually an error message
              + regular entry
              : non-important entry
              # tracer entry
              * network-related entry
              ~ zmodem/zedzap protocol error message
              = file transfer / mail session result message
              - console message only, not logged to disk
              <space> console message only, not logged to disk

```

```

DAY..... one or two digits
MONTH..... three-character month name
TIME..... 24-hr clock in the form hh:mm:ss

```

NOTES: (1) If the day is less than 10, there is a leading zero. (2) If the hour is less than 10, there is a leading space character, not a zero.

```

PROGRAM..... as many as 8 characters/digits, no spaces
MESSAGE..... as many as 80 characters/digits/etc
END-OF-LINE..... a CR/LF pair for MS-DOS

```

That comes to about 107 characters as the maximum line length. Each item is be separated by a space (not a TAB). Until you get to MESSAGE, none of the items can contain a space character ... that is the delimiter.

The tracer type statement is a "debug" sort of thing right now, but its real intent is to allow you to get extremely detailed information on a particular user (eg. "Trace Bri..."). That's in the future, though ... (Opus 2.00?).



## 11.3 STANDARD LINE ITEMS

There are three line items that are standard:

```
EMPTY LINE..... A program first sends an END-OF-LINE to the log
                    file to insure a fresh start for its entries.

BEGIN TIME..... The second line marks the start-up time. It has
                  a '+' symbol. The message itself is not defined,
                  but the item following PROGRAM must be the word
                  "begin" (case is irrelevant).

                  + 20 Sep 11:30:05 OPUS begin, v1.0, task=0
                  + 17 Jun  0:17:25 OPUS Begin, 910617 v1.70, Task=2

END TIME..... The last item is an off-line message. It is just
               like the BEGIN TIME, but represents the time
               when the program stopped execution. After
               PROGRAM you will find the word "end".

               + 17 Jun  4:00:00 OPUS End (84+0=84)
```

For Opus, the BEGIN line will include the version number and the current task number for multitasking environments. For Opus V1.70 onwards, it also includes the full date in the form YYMMDD which should allow for easy sorting by those so inclined.

For Opus, the PROGRAM name will always be "OPUS".

For Opus, the END line will always include the DOS errorlevel with which Opus exited (plus any errorlevel offset taken from the event file for the current behaviour window). The errorlevel offset is always indicated even when there is none, in which case it simply logs +0. For example, + 14 Jul 0:20:28 OPUS End (83+0=83)

## 11.4 LEVELS OF LOGGING

Not all logs are created equal. In Opus, there are several levels of logging: Trace, Verbose, Terse and a combined Trace and Verbose for masochists with large hard disks. The level is set by each individual sysop using a statement in the Opus control file (refer to the section 2 of this manual).

Here is a chart of the kinds of messages that appear in the various flavours of log:

	!	-	~	#	:	+	=	*	" "
TRACE	L		L	L		L	L	L	
VERBOSE	L				L	L	L	L	
TERSE	L								X
VIDEO	K	V		S	S	S	S	S	S

LEGEND: L = logged to disk  
V = always logged to video  
K = logged to video only if KEYBOARD mode is OFF  
S = logged to video only if SNOOP mode is OFF  
X = logged to video only if SNOOP & KEYBOARD modes are OFF



## 11.5 OOMP LOG MESSAGES

The following is an explanation of the log messages generated by the internal Opus echomail processing system:

<b>A={string}</b>	Echo_Name (one word, if several, separated by underscore).
<b>M={number}</b>	Number of copies made, should be number of new messages made times the number of boards you scan to, -1 if you're re-packing mail, (because whoever sent it to you has already seen it).
<b>C={number}</b>	Number of messages copied. If you have 30 messages, and 10 new ones are scanned and copied, the Count will be 10 messages.
<b>D={number}</b>	Messages already done, ie, if an area has 400 messages, 10 new ones arrive, 20 get scanned out, but 390 of that 400 were already scanned out some time previous.
<b>S={number}</b>	Number of bad messages skipped.

For example: A=COOKING M=1 C=0 D=1 S=0

## 11.6 OPUS LOG MESSAGES

The following are a selection of Opus log messages which you may encounter. Each log entry is briefly explained. Note: not all these entries will appear in every log, the entries below come from logs created using the LOG TRACE and LOG VERBOSE combination setting.

```
# 09 Jun 0:48:45 OPUS Password-protected session
```

Opus has called, or been called by, another mailer and is conducting a password protected mail session to exchange mail or files. The password is embedded in the nodelist data files (refer to the documentation that comes with your favourite nodelist compiler program).

```
: 09 Jun 0:48:45 OPUS Responding to: 3:711/909.0
: 09 Jun 0:48:45 OPUS Responding as: 3:711/401.0
```

As Opus V1.70 uses a "best match" approach based on the calling or called system's network address for determining which of its addresses (if more than one is defined) to use when conducting a mail session with other system's mailer, this information is logged for debugging purposes (refer to the explanation of the ADDRESS option in the Opus control file in section 2 of this manual for a full explanation).

```
* 09 Jun 0:48:47 OPUS Received: diddleysquat
```

Your system did not receive any mail or files from the other system during the mail session.

```
# 09 Jun 0:48:48 OPUS Nothing to send from 3:711/401.0 to 3:711/909.0
```

Your system had nothing to send to the calling or called system.

```
* 09 Jun 0:48:48 OPUS Connect: 0:01
```

How long the mail session lasted in Hours:Minutes format.

```
: 09 Jun 2:23:30 OPUS Inactive
+ 09 Jun 2:23:30 OPUS John Walker off-line. Calls=32, Len=54 Today=54
```

The caller to your system did not send any data (ie did not press a key) for 6 minutes. Opus logs this fact and disconnects (hangs up).

Calls	Total calls to your system
Len	Total minutes online this session
Today	Total minutes online so far today

```
# 09 Jun 6:39:50 OPUS Flagging C:\Opus\Outbound\005a00c8.Sa1 as sent
```

The specified mail was successfully sent, and now the archive file has been truncated to zero bytes.

```
: 09 Jun 6:41:16 OPUS Throughput = 14515 bps (151)
```

The speed of a mail or file transfer measured in bits per second (divide this figure by 10 to determine the CPS or characters per second). The figure in parentheses is the efficiency of the transfer measured as percentage of the CONNECT speed and the actual speed of the transfer. The above entry was the result of a mail session between two TrailBlazer PEP modems, Opus interpreted the CONNECT FAST connect message as a 9600 bps connect, so 14515 bps is equal to 151% of 9600 bps.

```
: 09 Jun 9:47:02 OPUS External show.bat
: 09 Jun 9:47:16 OPUS Return from show.bat (0)
```

Opus logs that an external program was run and that Opus then regained control of the system. In this case the external was a batch file. The figure in parentheses indicates the DOS errorlevel returned by the external program.

```
! 09 Jun 11:22:23 OPUS Expiration Date Warning
```

Opus logs the fact that the caller received the expiration date warning message which you defined in your Opus control file (refer to section 2 of this manual). The Opus caller account expiration system is explained in more detail in the Opus Operations Manual.

```
! 09 Jun 11:33:02 OPUS Z-SyncErr: CARRIER
~ 09 Jun 11:33:02 OPUS Cancelled
```

Opus detected a loss of modem carrier during the file transfer or mail session and cancelled the transfer or session.

```
: 09 Jun 11:39:11 OPUS OPED Max_Lines = 100
```

Opus logs the fact that when a caller chose the Opus full screen editor to enter a message, there was sufficient memory available to allow the caller to enter 100 lines in the message (refer to section 2 of this manual for details on setting the maximum number of lines allowed for a message in the OPus control file).

```
! 09 Jun 11:55:56 OPUS Kingsley Hancock isn't in user list
```

Kingsley Hancock logged on, but Opus could find no caller by that name in the user file. Opus then asked him whether he wanted to logon as a new user ...

```
+ 09 Jun 11:57:13 OPUS Kingsley Hancock calling
```

... and he wisely decided to do so, so Opus logs this fact.

```
: 09 Jun 11:57:52 OPUS ^OS = C:\OPUS\MISC\MEMSHP.BBS
```

Opus logs the fact that an Opus Embedded Command (OEC) was used to show the file MEMSHP.BBS to a caller (refer to section 4 of this manual for a full list of OECs).

```
: 09 Jun 11:57:52 OPUS ^OS = C:\OPUS\MISC\MEMSH.P.BBS
* 09 Jun 14:02:03 OPUS FiReq for `BOSS215*.*'
: 09 Jun 14:02:18 OPUS Throughput = 1513 bps (63)
= 09 Jun 14:02:18 OPUS DL-Z R:\Opus\misc\401_inf.arc 1059
```

Your system received a file request from another system for the file BOSS215\*.\* -- the other sysop used wildcards, especially for the file extension because he did not know the method of compression used to store the file on your system.

```
+ 09 Jun 16:25:38 OPUS Bev Freed calling
! 09 Jun 16:25:40 OPUS PwdErr: `Help'
! 09 Jun 16:25:42 OPUS PwdErr: `Help'
! 09 Jun 16:25:45 OPUS PwdErr: `Help'
! 09 Jun 16:25:46 OPUS PwdErr: `Help'
! 09 Jun 16:25:49 OPUS PwdErr: `Help'
! 09 Jun 16:25:49 OPUS +INVALID PASSWORD
```

Poor Bev, she seems to have forgotten her password AGAIN. Opus logs the fact that the password given did not match the password stored for this caller in the Opus user file. After the fifth unsuccessful attempt, Opus hangs up. This is a security measure; if there were lots of entries like this for one user, it may indicate that another caller was attempting to impersonate that user ... or that the caller has as bad a memory as Bev!

```
: 09 Jun 16:32:18 OPUS Reloaded R:\Opus\LastUs00.Dat (0)
+ 09 Jun 16:32:19 OPUS Relogging Trev Roydhouse
: 09 Jun 16:32:19 OPUS After RELOG User = Trev Roydhouse
```

Opus reloaded the details of the last caller from the LASTUS##.DAT file and relogged the caller using the SECURE and RELOG mechanisms (refer to section 2 of this manual for details of these options).

```
* 09 Jun 17:11:52 OPUS FiReq for `files'
~ 09 Jun 17:13:25 OPUS Kbd.<esc>
~ 09 Jun 17:13:25 OPUS Cancelled
* 09 Jun 17:13:25 OPUS File(s) not sent
```

Your system received a file request for your filelist of files for request. For some reason you pressed the ESCape key on your system and cancelled the request. (I was testing the system, not being nasty, to get this log entry for you.) Opus logs the fact along with a message that the file(s) was/were not sent.

```
: 09 Jun 17:15:34 OPUS Making file request
: 09 Jun 17:15:36 OPUS Throughput = 35 bps (1)
= 09 Jun 17:15:36 OPUS DL-Z R:\Opus\Outbound\02c701f5.REQ 7
~ 09 Jun 17:16:26 OPUS Kbd.<esc>
* 09 Jun 17:16:27 OPUS Partial file (Filelist.Arc)
```

Your system was making a file request from another system. The file containing the request(s) was transferred, but you decided to cancel the request using the ESCape key when you noticed the other system's filelist was 250mb long! Opus logs the fact the you received only a partial copy of the requested file. You may need to delete this file, which will be in your inbound files directory as BADWAZOO.nnn, along with the corresponding \*.Z marker file in your outbound mail holding area (refer to section 5 of this manual for details of these files).



```
# 09 Jun 17:28:36 OPUS MNP filter
```

This is how Wynn Wagner III described this log entry in September 1988 (yeah, ok, I admit it, I hoarded Wynn's messages from the MEADOW):

"When a session first begins, Opus-CBCS looks for the tell-tale signs of an MNP protocol. MNP is the error detection/correction/compression method begun by Microcom, and it's the most widely used method in generally available modems. All 9600 baud modems have some kind of error control method ... some 2400 baud modems do. MNP is just the most widely used.

"An MNP modem will spew out some bytes ta the start of a call. Some of those bytes have the potential of confusing Opus into thinking there's a netmail session. That's just one example of the problems MNP can cause. So, Opus tries to protect itself by being sensitive to MNP.

"If you have an MNP modem, the modem should use those bytes and NOT pass them on to Opus. In theory, if you have an MNP modem you should never see "MNP filter" in your log. That's just theory: in practice you probably will see it from time to time.

"The filter itself just waits for half a second (or so) then clears the inbound modem buffer. In effect, it fires a howitzer at the MNP material then goes on about its business.

"It's harmless if you see the message and harmless if you don't. The message is No Big Deal. 'So why is it logged?' you ask. Opus has a zillion little things like that filter. For example, the Opus command line has a small noise filter which is never discussed and never logged. As I recall, somebody suggested we needed a log entry for MNP. I don't recall why."

```
! 09 Jun 17:32:51 OPUS Can't open C:\Opus\misc\filelist.arc:
No such file or directory
```

Ooops, someone deleted the filelist to be sent to other systems which request the FILES filelist from your system.

```
+ 10 Jun 0:01:00 OPUS HouseCleaning
```

Opus executed the House-cleaning event (cleared the outbound holding area of the \*.?\*\$ files; refer to section 5 of this manual and the section of the Opus Operations Manual dealing with the Opus Event System for further explanation).

```
= 08 Jul 15:08:29 OPUS UL-Z F:\Mailfile\In\Badwazoo.001 34354
* 08 Jul 15:08:30 OPUS Finished partial file (What.exe)
```

Opus has logged the fact that this was the continuation of a zmodem/zedzap mail transfer which was not successfully completed on the first call (hence the Badwazoo file) but which has now been completed successfully.

```
: 08 Jul 18:44:32 OPUS Dirty word filter
```

Opus logs the fact that a caller attempted to log on using a name or combination of characters which you have placed in the Opus "trashcan" or "badname" file (refer to the USES NAMEFILTER and USES BADNAME options in the section2 of this manual).

```
: 08 Jul 20:58:47 OPUS User's last time Sun Jul 07 22:02:46 1991
: 08 Jul 20:58:47 OPUS Time/DL Zeroed
```

The DAILY time and download limits are zeroed IF a user is a Sysop or if the user has not called since midnight last night. The DAILY limits are zeroed, not the TOTAL limits.



```
* 08 Jul 18:57:31 OPUS FiReq for `OPUS_API.Z15'
: 08 Jul 18:57:39 OPUS ReqLimit = 30 i = 0
```

Opus logs the ReqLimit line indicating that file requests during this matrix behaviour window are limited to 30 minutes, the i= signifies how much of the time has already been used.

```
# 01 Jun 12:12:30 Connect 2400
* 01 Jun 12:12:33 AUG*MAC*BBS (3:711/911.0)
! 01 Jun 12:12:34 OPUS PwdErr: ``
# 01 Jun 12:45:23 Ring
```

Opus logs the fact that it expected a mail session password from the calling mailer and that the calling mailer failed to offer any mail session password. Opus hangs up in this situation. If the calling mailer offered an incorrect password, it would be logged (eg PwdErr: `wrong').

```
! 09 Jul 14:23:54 OPUS DENIED ACCESS: File area 31 from area 30
```

A caller tried to access file area 31 to which the caller did not have access. The caller was in area 30 at the time.

```
! 02 Jun 13:43:23 OPUS Missing node 119/567
```

Opus logs the fact that it could not find 119/587 in the nodelist when it attempted to call the system to deliver out-bound mail.

```
! 08 Jun 00:00:28 Can't lower DTR
```

Opus logs the fact that it was unable to lower the modem's Data Terminal Ready signal -- DTR may be forced high by your modem and may not have been set to follow CD, consult your modem manual.

```
! 23 Jun 13:46:43 OPUS Task 1: System crash detected
! 23 Jun 13:46:43 OPUS Last caller: John Walker (Jun 23 13:12:55)
```

When Opus task number 1 was restarted it found an ACTIVE01.DAT file, and as Opus is house-trained, it knew it would not have left that file lying around if it had not been sick, so it logs this fact.

```
! 27 Jun 18:00:00 Expiration Date Priv Changed
```

A caller has passed the expiration date the sysop set in the caller's user record in the Opus user file, so Opus logs the fact that it is reducing the caller's access to the level specified by the sysop in the Opus control file (refer to the EXPIRE DAYS and EXPIRE PRIV options in section 2 of this manual).

```
: 27 Jun 2:05:43 OPUS Scanning MEADOW to 54/99
! 27 Jun 2:05:44 OPUS Too many SEEN-BYs
```

Another system sent you an echomail message with more than 250 systems listed in the SEEN-BY lines!



**12****USING DESQVIEW**

DESQview by QuarterDeck Systems is a commercial product that provides the ability to run multiple DOS programs simultaneously. It is very popular and successful for use on 80386 IBM compatible systems for running Opus concurrently with other programs. This allows the sysop of the more powerful 80386 systems to perform CBCS maintenance tasks and run other programs concurrently while Opus continues to operate. There are several important aspects in the successful operation of Opus under DESQview. They are described in the following sections.

DESQview can be utilized on the 80xx and 80286 processors, but its function is limited, as the amount of memory remaining and overhead incurred generally results in less than satisfactory results. For these reasons, discussion is limited to the application on an 80386 machine.

**12.1 FOSSIL DRIVERS AND DESQVIEW**

Three fossil drivers exist for the IBM compatible system. These are Opuscomm (by Bob Hartman), BNU, and X00 (by Ray Gwinn). All three fossil drivers have been used successfully for running Opus under DESQview. There are a few basic rules to follow.

- Load the fossil driver BEFORE starting DESQview
- Load other fossil dependent programs AFTER loading the fossil driver, but BEFORE starting DESQview. Example programs are VFOS\_IBM, Binkleyterm's video fossil driver, and CHAT.COM, Opus's chat and yell module.
- After starting DESQview, open the Opus window to start Opus. The first program to be run should be the appropriate capture program that comes with the fossil driver you are using.

A National Semiconductors NS16550AFN UART chip is virtually required to run communications under DESQview with high speed modems. This chip has special buffering. In order to use it, you will need to replace the existing UART chip (usually an 8250 or 16450) with the NS16550AFN (this costs around \$US15 to \$A50). However, you can only do this if the chip on your motherboard, serial i/o card or multi function card is socketed. You will often find that cards have one chip soldered in for COM1:, but a socket for a second UART to be used for COM2:. In these cases you usually also need to purchase an additional 1488 and an additional 1489 chip (these cost only a few cents). Consult your card or motherboard's instruction manual. If the worst comes to pass, you may need to purchase a separate serial I/O board, and install the chip yourself. Be sure that the serial I/O or multifunction card you purchase is socketed to make the installation of the chip easy. NS16550AFN chips are available from most electronic parts stores or mail order houses.

Most users of DESQview on an 80386, also use Quarterdeck's excellent expanded memory manager called QEMM. This manager allows memory resident programs and drivers to be loaded into "high" memory in the address range between 640k and 1024k, thus leaving more conventional memory (0-640k) available to other programs. Some fossils will not work when loaded high on some brands of machines. X00 contains special code to make it more adaptable to loading high. The best advice here is to get everything working without loading high. Then come back and try to load the fossil driver high. A typical symptom of incompatibility with loading high is machine lockup during boot up. Frequent Memory Exception #13 errors may be cured by increasing the number of DOS buffers loaded low in conventional memory.

**12.2 DESQVIEW WINDOW SETUP**

A typical setup for Opus under DESQview 2.34 is shown here:

```
Memory Size (in K): 350
Program...: c:\opus\nerf.bat
Parameters: (blank)
Directory.: c:\opus
Writes text directly to screen.....: [N]
Displays graphics information.....: [N]
```



```
Virtualize text/graphics (Y,N,T).....: [N]
Uses serial ports (Y,N,1,2).....: [Y]
Requires floppy diskette.....: [N]
System Memory (in K).....: 0
Script Buffer Size.....: 1000
Maximum Program Memory Size (in K)...: (blank)
Maximum Expanded Memory Size (in K): (blank)
Text Pages: 1
Graphics Pages: 0
Initial Mode: (blank)
Interrupts: 00 to FF
Close on exit .....: [N]
Allow Close Window command.....: [N]
Uses math coprocessor.....: [N]
Share CPU when foreground.....: [Y]
Can be swapped out .....: [N]
Uses its own colors.....: [Y]
Runs in background .....: [Y]
Keyboard conflict (0-F).....: [0]
Share EGA when foreground/zoomed.: [Y]
Protection level (0-3).....: [0]
```

The "Writes text directly to screen" and "Virtualize text/graphics" responses can be set to "N" if Opus is run in Video IBM mode and no external programs use direct video writes.

The option "uses serial ports" can be set to [N], if you do not have any external utilities which use the serial port directly. Since Opus does all I/O via the fossil driver, it does not use the serial port directly. If you use any external file transfer protocols that do not use the fossil driver, be sure to answer [Y] to this question.

---

## 12.3 TUNING

Tuning is the setting of the foreground/background ticks in the DESQview setup menu called "Performance". The setting will depend on the type of processor you have, how fast it is, whether you have memory caching and several other factors. The best advice is to experiment with the performance settings. Setting the foreground and background ticks equal is advisable, if your processor can handle it. For example, a 386/25 might run very well with foreground set to 2 ticks and background set to 2 ticks. If you use your PC for lots of other tasks, Opus will typically be running in the background. On slower processors (386/16), you may need to weight the background more heavily to insure that communications continue to function while you work in the foreground. A typical setting for a 386/16 might be foreground set to 3 ticks and background set to 8 ticks.

---

## 12.4 FILE SHARING AND SEPARATION

Opus has some features in its design which allow you to run multiple copies in separate DESQview windows. The primary requirement to run multiple copies of Opus is to insure that two copies of Opus cannot write to the same file concurrently. This means that you must either have some form of file locking or utilize a separate copy of each read-write support file for each copy of Opus running. You do not need to separate support files which are read-only.

### 12.4.1 USER FILE

Opus supports user file locking via DOS's SHARE.EXE program. In order to utilize it, load SHARE before starting DESQview and enable locking in the Opus control file by uncommenting the statement:

```
LOCKS
```

and recompiling the control file with NACL.

### 12.4.2 TASK NUMBERS



Opus keeps various support file separate by the use of a task number. Each running copy of opus should have its own control file and have a unique task number. For example, if you have two copies of Opus to run under separate DESQview windows, assign the first copy as task=1 and the second as task=2. These task numbers are utilized internally by Opus to point to filenames. See the explanation of OEC files and task numbering conventions in section 4.2 of this manual.

### 12.4.3 BITS AND PIECES

You should define a separate control file for each copy of Opus that you run. You should also specify separate pathnames for log files and schedule files, as Opus does not have any special provision to share these files with other programs.

When Opus is processing mail, it creates a flag file called INMAIL##.\$\$\$ , where ## is the hexadecimal task number. This flag file tells other copies of Opus that mail processing is in progress and prevents more than one copy of Opus processing mail at a given time. Simultaneous mail processing could be disastrous. This flag file is deleted by Opus after internal processing is complete. If Opus is set to exit after internal processing, the flag file will remain on the disk for use to communicate to external mail processing programs that mail has been received and by which copy of Opus.

Opus was designed to minimize, but not totally eliminate all possible conflicts between multiple copies for simultaneous mail processing. It is highly recommended that you only allow one copy of Opus do the actual mail processing.



# 13

# USING DOUBLEDOS

DoubleDOS by SoftLogic Solutions is a commercial product that provides the ability to run two DOS programs simultaneously. It is very popular and successful for use on 8086, 8088, 80286 and 80386 IBM compatible systems for running Opus concurrently with other programs. In fact, it works better than DESQview on 80286 and below systems, although you will get somewhat less functionality compared with DESQview. It comes down to a tradeoff of cost and utility versus the hardware you have to work with. A DoubleDOS setup will allow you to perform CBCS maintenance tasks and run other programs concurrently while Opus continues to operate. Some sysops have successfully used DoubleDOS to run Opus CBCS systems on the one computer. There are several important aspects to the successful operation of Opus under DoubleDOS. They are described in the following sections.

---

## 13.1 FOSSIL DRIVERS AND DOUBLEDOS

Three fossil drivers exist for the IBM compatible system. These are Opuscomm, BNU, and X00. The Opuscomm and X00 fossil drivers have been used successfully for running Opus under DoubleDOS. Although BNU did not exist at the time (circa 1989) when I (Trev Roydhouse) ran Opus under DoubleDOS, but it should work just as well.

If you are using the Opuscomm, BNU or X00 fossil programs as Terminate and Stay Resident programs, run the relevant fossil program in the DoubleDOS partition in which you are going to run Opus.

If you are using the X00 device driver version of the fossil, loading the fossil device driver using your DDCONFIG.SYS file rather than CONFIG.SYS will save you up to 14K of memory for the your two DOS partitions (see the DDCONFIG.SYS sample control file in the DoubleDOS Setup section below).

After starting DoubleDOS, the Opus partition(s) should have the appropriate "capture" program that comes with the fossil driver you are using run to make sure that the fossil driver has control of the communications interrupt.

A National Semiconductors NS16550AFN UART chip is virtually required to run communications under DoubleDOS with high speed modems. This chip has special buffering. In order to use it, you will need to replace the existing UART chip (usually an 8250 or 16450) with the NS16550AFN (this costs around \$US15 to \$A50). However, you can only do this if the chip on your motherboard, serial i/o card or multi function card is socketed. You will often find that cards have one chip soldered in for COM1:, but a socket for a second UART to be used for COM2:. In these cases you usually also need to purchase an additional 1488 and an additional 1489 chip (these cost only a few cents). Consult your card or motherboard's instruction manual. If the worst comes to pass, you may need to purchase a separate serial I/O board, and install the chip yourself. Be sure that the serial I/O or multifunction card you purchase is socketed to make the installation of the chip easy. NS16550AFN chips are available from most electronic parts stores or mail order houses.

---

## 13.2 MEMORY CONSIDERATIONS

Memory considerations vary depending on whether you are using DoubleDOS V4.00 or the later DoubleDOS V5.00 which gave me two DOS partitions of 553K and 408K ... how? Read on.

The BUFFERS= statement in your CONFIG.SYS file should be set to at least 10 even if you are running a disk cache. Specifying too few buffers, used to cause some very bizarre side effects that took many hours to track down.

### 13.2.1 DOUBLEDOS VERSION 4.00

Under DoubleDOS V4.00 the size of each of your two DOS partitions is limited by the size of your computer's conventional memory, less approximately 20-40K for DoubleDOS itself, and then two lots of 25-40K for room to load the DOS shell program (COMMAND.COM) in each partition.



For further memory saving hints, refer to the comments in the sample DDCONFIG.SYS file in the section on DoubleDOS Setup below.

### 13.2.2 DOUBLEDOS VERSION 5.00

Under DoubleDOS V5.00 the size of each of your two DOS partitions is not limited by the size of your computer's conventional memory!

As usual, there are a couple of gotchas ...

- Your computer must have EXPANDED memory and your expanded memory board must be either EEMS compatible or EMS/LIM V4.00 compatible in hardware. Some of the early Intel Above Board memory boards have EMS/LIM V4.00 software driver compatibility, but not hardware compatibility. The AST range of memory boards have always had EEMS compatibility, but you must also be using an EMS V4.00 software driver (AST's REMM.SYS V4.00 or above -- available from many Bulletin Boards); and
- You must be able to disable your motherboard conventional memory to 256K and then backfill memory using the memory on your expanded memory card (the AST range can do this).

Given that your system fulfills the above two essential requirements, you will be able to take advantage of expanded memory to give you two 400K+ DOS partitions.

To get the 553K and 448K DOS partitions I mentioned earlier, you need to acquire a copy of the program called ADDRAM V1.0 by Mart Del Vecchio. It is free for personal use and comes with full assembly source code for the curious hacker. To use ADDRAM:

- Your expanded memory board must be able to address memory in the address range between A000 and B000 (all of the AST memory boards can do this - the Intel Above Board Plus cannot do this); and
- There must be unused memory starting at A000 (ie you must not be using an EGA or VGA video card).

When run in a V5.00 DoubleDOS partition, ADDRAM can add an extra 64K (if you have a mono video card) or 96K (if you have a CGA video card) to each DoubleDOS memory partition.

After DoubleDOS has loaded COMMAND.COM in each partition, you then have a TOP partition of 515K and a Bottom partition of 408K.

ADDRAM comes with excellent documentation, and a helpful program EMSDATA.EXE which will tell you whether your expanded memory board can be used with ADDRAM. Refer to that documentation when trying to set up your system to use ADDRAM.

---

## 13.3 DOUBLEDOS SETUP

Set out below is the commented DDCONFIG.SYS file which I used with DoubleDOS V4.00 and V5.00 to run Opus in one partition while I did BBS maintenance or wordprocessing in the other:

```
***           DoubleDOS Options Control File           ***

;place a * at the beginning of a line to de-select an option
;remove the * to select an option

menu = line1                      ;put short menu on line 1 (1-48)
                                   ;this option saves you 9K of memory

display = mono                    ;use monochrome adapter only
display = text                    ;no graphics, saves 16K memory
```



```
* print buffer = 2           ;comment out to save more memory

print driver = direct       ;use direct drive no buffer reserved

* hot key = ALT              ;key to activate keys below for
hot key = CTRL               ;special functions

                        ;The keys below are used WITH the HOT KEY

menu key = 83                ;DEL key calls DoubleDOS menu
exchange key = 1             ;ESC key toggles tasks on/off screen
suspend key = 58             ;Caps Lock key suspends / unsuspends
                                ;INVISIBLE program
clear key = 14               ;Backspace key clears contents of
                                ;the keyboard type ahead buffer for
                                ;the VISIBLE program

top device = X00.SYS E B,1,19200 ;loading X00 here rather than in
                                ;CONFIG.SYS saves 14K memory
top device = DDCOMIO /2      ;see the OUTSIDE PROGRAM section
                                ;of the documentation below

top program = addram /4      ;startup commands to top memory section
top program = cd c:\opus     ;change to Opus directory
top program = xu capture on  ;capture the comms interrupt for X00
top program = nerf           ;run the Opus batch file

bottom program = addram1 /4 ;startup commands for bottom memory

* max section = top          ;max partition size (515K) top or bottom

priority = equal             ;50% priority to Opus

com1 = bottom                ;assign com1 to bottom partition
com2 = top                   ;assign com2 to top Opus partition
                                ;NB com2 has higher priority than com1

; refer to the DoubleDOS manual for more configuration
; information and options
```

---

## 13.4 OUTSIDE PROGRAMS

There is an annoying problem with some Opus "outside" or external programs (eg simple games, questionnaire programs etc); they freeze the non-Opus partition while they are running. This can be very annoying. It was so annoying that I took time out in 1989 to solve the problem. Unfortunately I no longer remember what caused the problem, so I cannot tell you.

I wrote a DoubleDOS-aware replacement for GATEWAY (used to redirect screen i/o over the modem port with many outside programs that are/were not fossil aware). The device driver solves the "freezing" problem and only takes up a couple of hundred bytes. If you find you need a copy, it can be file requested from 3:711/401 (PEP) or 3:711/501 (V32) under the magic name DDCOMIO. You are unlikely to find it anywhere else because there was never any demand to publicly release it.

---

## 13.5 PERFORMANCE

Do not forget to specify in the Opus control file that you are using the DoubleDOS multitasker program (refer to the MULTITASKER option in section 2 of this manual). Specifying this fact will cause Opus to release its unused time slices to the other DoubleDOS partition.

Opus V1.03x ran well under DoubleDOS V4.0x and V5.00 on my 10mhz 80286 AT clone driving an internal Telebit TrailBlazer modem with the fossil locked at 19,200 bps. The priority was set to give each partition EQUAL priority.

---

## 13.6 FILE SHARING AND SEPARATION

Opus has some features in its design which allow you to run multiple copies in separate DoubleDOS partitions. The primary requirement to run multiple copies of Opus is to insure that two copies of Opus cannot write to the same file concurrently. This means that, in theory, you must either have some form of file locking or utilize a separate copy of each read-write support file for each copy of Opus running. You do not need to separate support files which are read-only. In practice, there were never any problems provided that each copy of Opus had a separate log file.

### 13.6.1 USER FILE

Opus supports user file locking via DOS's SHARE.EXE program. In order to utilize it, load SHARE before starting DoubleDOS and enable locking in the Opus control file by uncommenting the statement:

```
LOCKS
```

and recompiling the control file with NACL.

In practice, you probably will not need to use SHARE; it has many problems aside from chewing up valuable memory and reducing the size of your two DoubleDOS partitions.

### 13.6.2 TASK NUMBERS

Opus keeps various support file separate by the use of a task number. Each running copy of opus should have its own control file and have a unique task number. For example, if you have two copies of Opus to running, assign the first copy as task=1 and the second as task=2. These task numbers are utilized internally by Opus to point to filenames. See the explanation of OEC files and task numbering conventions in section 4.2 of this manual.

### 13.6.3 BITS AND PIECES

You should define a separate control file for each copy of Opus that you run. You should also specify separate pathnames for log files and schedule files, as Opus does not have any special provision to share these files with other programs.

When Opus is processing mail, it creates a flag file called INMAIL##.\$\$\$ (provided you have told Opus that you are running DoubleDOS), where ## is the hexadecimal task number. This flag file tells other copies of Opus that mail processing is in progress and prevents more than one copy of Opus processing mail at a given time. Simultaneous mail processing could be disastrous. This flag file is deleted by Opus after internal processing is complete. If Opus is set to exit after internal processing, the flag file will remain on the disk for use to communicate to external mail processing programs that mail has been received and by which copy of Opus.

Opus was designed to minimize, but not totally eliminate all possible conflicts between multiple copies for simultaneous mail processing. It is highly recommended that you only allow one copy of Opus do the actual mail processing.

**14****OPUS UTILITIES**

The heritage of the Opus CBCS is one of incredible support from many good software authors. Opus was designed from the outset to provide hooks and open information to allow additional programs to be integrated for expanding its functionality and ease of maintenance. Even before the release of Opus CBCS V1.70, a programmer's toolkit including necessary data structures, documentation and sample source code was made available to make it even easier for the utility author to write support programs for Opus.

Unfortunately it is not feasible to include a list of available Opus utilities in this manual because such list would quickly be out of date. Instead, the following information should help you to find the latest Opus utilities:

- There is an Opus Archive node. This node is listed in the Opus help screen when you press the ENTER key at the ready prompt. The Opus Archive will have a complete list of utilities and specifications available for Opus CBCS.
- There are also official OpusInfo help nodes located in the USA, Australia and Holland. These nodes, which are listed in the Opus help screen when you press the enter key at the ready prompt, usually carry a good selection of the latest Opus utilities.

Another source of information on the latest Opus utilities is the FidoNet MEADOW echomail conference which is open to ALL sysops of Opus CBCS systems.

**14.1 CATALOG OF OPUS UTILITIES****Opus Catalog of Amazing Things**

Copyright 1991 by Keith Ford  
in accordance with the Opus philosophy

Dedicated to everyone who has helped someone

"OCAT can be used to help the Opus sysop locate software and other files that will make life "easier?" I quoted easier because the same sysop may find other neat "programs" to use and end up making his/her system more complex. But that's okay!

"OCAT can serve as a registry for filenames and program names. No sense in sending out a new utility if the sysop will have to change the name you created since he already has a file by than name.

"This is a template for OCAT. To have an entry added to OCAT, please fill out a copy of this template with the proper information and mail it to one of the following:

Fidonet : OCAT @ 1:373/12  
Usenet : ocat@umagic.fidonet.org  
BBS : +1 205 830 2362  
Postal : 203 Creek Trail, Madison, AL 35758

"Please note that the editor reserves the right to make minor changes  
TO your text to coordinate it with the entire database."

-- Keith Ford.

-----  
Program Name : ( 1)  
Version : ( 2)



Opus Version : ( 3)  
Release Date : ( 4)  
Release Filename : ( 5)  
Software Class : ( 6)

Author : ( 7)  
Fidonet : ( 8)  
Other : ( 9)

Keywords : (10)

Description : (11)

-----  
Description of items listed in the template.

- 1 Name BY WHICH THE program is known. If your program name is an acronym, you should use what is most common. For example, OBUL is Opus bulletin utilization language, but it is most commonly called OBUL (oh bull :-). Other authors should avoid conflicting names.
- 2 Version number of program for this description. Use 'none' if appropriate.
- 3 Version of Opus this program supports. (1.0x, 1.1x, 1.2x, any, etc.)
- 4 Date author released this version.
- 5 Filename, without extension, author used for released program. Other authors should avoid conflicting names.
- 6 Type of software:
  - 7 Commercial Public Domain (all rights released) Freeware (no fee, some rights reserved) Shareware (fee for continued usage) Crippleware (fee required to enable all features) Nagware (fee required to turn off annoying screens)
- 8 The last two are supposed forms of shareware which have obtained these nicknames from the industry and users. XLAXNODE is an example of crippleware as the program cannot be run unattended from batch until you register. The latest SHEZ is an example of nagware as a message pops up each time you start it and requires a key hit to get past.
- 9 Author's name.
- 10 Author's fidonet address for correspondence.
- 11 Other addresses for author. (Usenet, US Mail, phone, etc.)
- 12 Words that categorize the program, useful for quick searches. Opus should not be one of them.
13. Paragraph description of programs function, usage, requirements, compatibility and such. Might also include where the program is available, if it has a frequest magic word, if source is included, and anything else that might be useful to readers.



# Index

—  
\_ECHO, 80  
\_FILE, 80  
\_GOODBYE, 81  
\_HISTORY, 80  
\_MAIL, 80  
\_MAIN, 80  
\_MAKE\_FILE\_MACRO, 81  
\_MAKE\_MACRO, 81  
\_MAKE\_MSG\_MACRO, 81  
\_MAKE\_SECTION, 80  
\_MESSAGE, 80  
\_MY\_MACRO, 81  
\_MY\_SECTION, 80  
\_OUTSIDE, 81, 107  
\_PICK\_SECTION, 80  
\_SECTION\_MODE, 80  
\_SEE\_CHAT, 81  
\_SEE\_LINES, 81  
\_SEND\_MESSAGE, 52, 80  
\_SETUP, 80  
\_SHOW, 81  
\_STATS, 81  
\_SYSOP, 80  
\_SYSOP\_MESSAGE, 80  
\_USERLIST, 81  
\_VERSION, 81  
\_YELL, 81

## 8

8250 UART, 14, 129

## A

ABOUT file, 32  
access  
  area, 61  
  edit, 64  
ACTIVE##.DAT, 8, 80  
ADDRESS, 134  
address  
  FidoNet, 18  
adopt, 87  
alias, 43, 83  
Amiga, 62  
ANSI, 82, 90  
application, 61  
ARC, 86

ARCE.COM, 19  
ARCmail, 19  
arcmail, 18  
area, 60  
  access, 61  
  name, 69  
  rules, 72, 84  
  section, 73  
  title, 73  
areacode, 119  
ARJ, 86  
ARJ.EXE, 19  
ARJmail, 19  
ARQ, 13  
ASCII, 12, 62, 82  
Asia, 62  
ass, 55  
asstsysop, 3  
AT modem commands, 16  
Australia, 1  
autoanswer, 13  
Avatar, 82, 90

## B

BADECHO, 28  
BADNAME, 49  
barricade, 62  
barrocade, 49  
baud, 14, 119  
  file transfer, 59  
  minimum, 27, 47  
  user, 36, 58  
BBS file format, 89  
Beebe, William, iii  
Binkley, 29  
Binkleyterm, 129  
BNU, 129, 133  
Boone, Doug, 1, 13, 45, 62  
bossnode, 20  
BRIEF editor, 42

## C

carrier, 119  
carrier detect, 16  
charset, 11, 62  
chat, 7, 59, 80  
CHAT##, 8  
CHAT.COM, 129  
CHG\_AREA, 84, 86  
Christensen, Ward, v  
city, 43  
clerk, 3  
clock, 40  
CM, 31

command-line switches, 105  
COMMAND.COM, 81, 121  
CONFHELP.BBS, 43  
CONFIG, 50  
CONFMENU, 50  
CONTENTSHELP, 50  
continuous mail, 31  
control-A, 27  
CRASH, 24, 26, 31  
crash mail, 21  
CTS, 17

## D

Data Terminal Ready, 14  
database, 9  
date format, 40  
Davis, Bob, 1  
DAYLIMIT, 51  
default log, 6  
DESVIEW, 7, 121, 129  
dial, 119  
diddleysquat, 123  
directory listing, 39  
disgrace, 3  
domain, 63  
doorway, 35  
DOS  
  ERRORLEVEL, 4  
  file, 3  
  path, 3  
dos, 119  
DoubleDOS, 7, 121, 133  
download  
  free, 65  
  limit, 37  
  log, 51  
  path, 63  
  protocol, 39  
  ratio, 37, 72  
  temporary path, 86  
DSR, 17  
DTR, 14  
dupes, 63

## E

Echo Guard, 21  
echomail, 18, 71  
  address, 61  
  inbound only, 65  
  passthrough, 69  
  points, 29  
  routing, 72  
  scan, 72  
  scanning, 21, 31  
  tag, 64

toss, 32  
 tossing, 20  
 ECHOTOSS.LOG, 23, 32  
 EconoScan, 21  
 ED\_ABORT, 87  
 ED\_ATTR, 87  
 ED\_CHG, 87  
 ED\_CONT, 87  
 ED\_DEL, 87  
 ED\_DISK, 87  
 ED\_FROM, 87  
 ED\_INSRT, 87  
 ED\_LIST, 87  
 ED\_SAVE, 87  
 ED\_SUBJECT, 87  
 ED\_TO, 87  
 ED\_UUCP, 87  
 edit, 19  
 edit access, 64  
 editor  
   local, 42  
 EID, 21  
 embedded commands, 89  
 English, 12  
 ENGLISH.MNU, 75  
 equipment, 13  
 error  
   list, 117  
 ERRORLEVEL, 4, 81  
 errorlevel, 113  
 Europe, 38  
 European date, 40  
 event, 114  
   forced, 48  
 EXIT, 81  
 expiration, 95  
 expiration date, 38  
 external mailer, 114  
 external protocol, 39  
 EXTMAIL, 22  
 extra, 3

## F

F\_ADOPT, 87  
 F\_ARCDIR, 86  
 F\_DNHANG, 86  
 F\_DNLD, 86  
 F\_DSPL, 86  
 F\_HELP, 65, 87  
 F\_HURL, 86  
 F\_KILL, 86  
 F\_LOCATE, 5, 86  
 F\_LONG, 86  
 F\_NEW, 86  
 F\_RAWDIR, 86  
 F\_TEMP, 86

F\_TITLES, 86  
 F\_UPLD, 86  
 favored, 3  
 FidoNet, 18, 30, 96  
 fidonet.org, 63  
 FIDUSER.LST, 33  
 file  
   area list, 51  
   date, 39  
   locking, 6  
   raw list, 39  
 file attach, 24  
 file list, 32  
 file specifier, 3  
 FILEAREAS, 51  
 FileReq  
   file list, 32  
   OKFILE, 33  
 FileReq ABOUT, 32  
 FILESBBBS.DAT, 86, 87, 95  
 FIXTOSS.EXE, 32  
 Ford, Keith, 137  
 format  
   date, 40  
 FOSSIL, 14, 36, 81  
 free, 65  
 Freed, Bev, 33, 125  
 French, 12, 62  
 FSC-0050, 11  
 FTB, 96  
 FTSC, 20  
 full-tilt boogie, 96

## G

GATEWAY, 135  
 GBS file format, 89  
 German, 62  
 goodbye, 76  
 goto, 119  
 Guido, 1, 9, 34  
 GUUCP, 23  
 Gwinn, Ray, 129

## H

hand hold, 44  
 handshaking, 17  
 Hartman, Bob, 129  
 Hayes, 16  
 Heckifiknow, 13  
 hidden, 3  
 HIDEPWD.EXE, 44  
 HISTORYMENU, 51  
 HOLD, 26  
 hold for pickup, 24

host, 29, 30, 31  
 host-routing, 31  
 HOTFLASH, 76, 82  
 housecleaning, 126  
 HST, 17  
 HST modem, 15  
 hub, 29, 30

## I

IBM, 62, 83, 90  
 inbound file path, 28  
 inbound message path, 28  
 inbound processing, 19  
 init, 119  
 INMAIL##.\$\$\$ , 8, 131, 136  
 INQUIREHELP, 51  
 INTERLINE, 52  
 InterNet, 63  
 ISO, 62  
 Italian, 62

## J

Japanese date, 40  
 Jennings, Tom, v

## K

Kermit, 39, 110  
 key, 3, 60  
   initial, 44  
 killsent, 25, 26  
 KUPLOAD, 66

## L

LAN, 7, 44  
 language, 11, 89  
 LASTREAD, 41  
 LASTUS##.DAT, 8, 48, 80, 108  
 Latin, 62  
 LEAVING, 52  
 level  
   menu, 75  
 LHA, 86  
 LHA.EXE, 19  
 limited, 3  
 line-oriented editor, 53  
 LISTSYSOP, 41  
 local, 67  
 local calls, 31  
 LOCATEHELP, 52



lock, 3, 60  
 menu, 75  
 log file, 6, 121  
 LOGO.BBS, 52, 89  
 logoff, 50  
 logoff messages, 42  
 LORE, 56, 79, 83, 87  
 LOREHELP, 53  
 LZH, 86  
 LZHmail, 19

## M

macro, 53, 81  
 mail  
   continuous, 31  
 MAIL ONLY mode, 22  
 MAILCHECK, 65  
 mailer  
   external, 23, 114  
 MAILER.BAT, 22  
 main menu, 53  
 MAKE, 53  
 matrix, 18, 68  
   script, 119  
 MAYBENEW, 54  
 Meadow, 64, 67, 96, 137  
 menu, 75  
   area, 68  
   custom, 50  
   history, 51  
   main, 53  
   picture, 43  
 message  
   area list, 54  
   in-transit, 29  
   local, 67  
   logoff, 42  
   private, 25, 70  
   renumbering, 65  
   size, 24, 68  
   upload, 74  
   UUCP, 26  
 MNP, 14, 126  
 modem  
   dialing prefix, 15  
   handshaking, 17  
   initialization, 16  
   ring, 13  
 Modem7, 110  
 monitor  
   height, 7  
   width, 7  
 More?, 82  
 mouse, 17  
 MSG\_CHILD, 84  
 MSG\_EDIT\_NEW, 84  
 MSG\_EDIT\_REPLY, 84  
 MSG\_FRWD, 85

MSG\_HELP, 65, 84  
 MSG\_HURL, 85  
 MSG\_INQ, 85  
 MSG\_KILL, 85  
 MSG\_LAST, 84  
 MSG\_LIST, 84  
 MSG\_NAREA, 84  
 MSG\_NEXT, 84  
 MSG\_NONSTOP, 84  
 MSG\_PARENT, 84  
 MSG\_PRIOR, 84  
 MSG\_QINQ, 84  
 MSG\_QUICK, 84  
 MSG\_RULES, 84  
 MSG\_SCAN, 6, 84  
 MSG\_UPLD\_NEW, 85  
 MSG\_UPLD\_REPLY, 85  
 MSG\_XPRT, 85  
 MSGAREAS, 54  
 MSGEXTERN, 67  
 MSGID, 21, 61, 63  
 MSGTMP.TMP, 42  
 multiline  
   chat, 7  
   common storage, 10  
 MULTITASKER, 135  
 multitasker, 7, 10

## N

NACL, 68, 75, 76, 77, 108, 110  
 name  
   BBS, 8  
   filter, 49  
   real, 43  
 NAMEFILTER, 55  
 NERF.BAT, 22  
 NETINFO path, 28  
 NEWUSER1.BBS, 55  
 NEWUSER2.BBS, 56  
 nickel  
   your, 29  
 Nilsson, Ulf, iii  
 noblesse oblige, 76  
 nodelist, 31  
   version 6, 34  
 NODELIST.DAT, 15  
 normal, 3  
 Norwegian, 62  
 NS16450 UART, 14  
 NS16550AFN UART, 14  
 NS16550AFN UART, 129, 133  
 nudity, 72  
 NUL log, 6

## O

OBUL, 138  
 OCAT, 137  
 OEC, 82, 89  
 OKFILE, 33  
 OPARSE.EXE, 31, 34  
 OpEd, 45, 56, 79, 83, 87  
 OPMED, 11  
 Opuscomm, 129, 133  
 Origin, 29  
 Origin line, 61, 69  
 OUFM, 81

## P

PAK, 86  
 ParseLst, 15  
 passthrough echomail, 69  
 password  
   file, 10  
   hide, 44  
 path, 8  
 path specifier, 3  
 pattern, 120  
 peek, private messages, 70  
 PEP modem, 15  
 phone, 120  
   charges, 13  
 PICTURE, 70  
 PKT file, 21  
 PKUNZIP.EXE, 19  
 point  
   address, 18  
   PointNet, 20, 29  
 pre-registration, 44, 46  
 PREFILES, 70  
 print, fine, ii  
 private message, 27  
 privil, 3  
 privilege levels, 3  
 protocol, 82, 110  
 protocol, file transfer, 39  
 PS-8, 62  
 PUPLOAD, 71

## Q

QEMM, 129  
 QMail, 29  
 QuarterDeck System, 129  
 questionnaire, 45, 83  
 quote-of-the-day, 56

**R**

RAM disk, 21  
 ratio, 72  
 REBOOT, 108  
 reboot, 9  
 refund, 48  
 RELOG, 35  
 relog, 48  
 RENUM, 65  
 REPLACEHELP, 56  
 request attribute, 25  
 RETURNING, 57  
 ring, 13  
 ROOKIE, 57  
 route, 72  
 Roydhouse, Trev, iii, 133  
 RTS, 17  
 rule, 72  
 rules, 84  
 RULES.BBS, 84  
 RUN, 81

**S**

S0 modem register, 13  
 scan, 31, 72  
 schedule, 114  
   file, 10  
 scientific date, 40  
 script, 119  
 SDOS, 81  
 SEAdog, 121  
 SeaLink, 110  
 section, 46, 47, 57, 73, 80  
 section specification, 4  
 SECTMENU, 57  
 security, 30, 60, 61, 125  
   barricade, 62  
   keys and locks, 3  
 SEEN-BY, 27, 61, 96  
   points, 29  
 session, 120  
 SET+ASKGRAPH, 82  
 SET\_ASKGRAPH, 82  
 SET\_BLOCK, 82  
 SET\_CITY, 83  
 SET\_CLS, 82  
 SET\_DEF, 82  
 SET\_EDIT, 83  
 SET\_GMENUS, 82  
 SET\_HELP, 82  
 SET\_IBM, 83  
 SET\_KEYS, 82  
 SET\_LANG, 83

SET\_LEN, 82  
 SET\_LIST, 83  
 SET\_MENU, 82  
 SET\_MORE, 82  
 SET\_NULLS, 82  
 SET\_PWD, 82  
 SET\_TABS, 82  
 SET\_TIME, 82  
 SET\_VID, 82  
 SET\_WIDTH, 82  
 SHARE.EXE, 6, 130, 136  
 shit, 55  
 SHOWTIME, 49  
 snoop, 9  
 SoftLogic Solutions, 133  
 software, snazzy, 96  
 Spanish, 62  
 speed, 120  
 SRUN, 81  
 stdaux, 5  
 stdprn, 5  
 Swedish, 62  
 SYS\_AREA, 85  
 SYS\_MAIL, 85  
 SYSFILE.DAT, 61  
 SYSMMSG.DAT, 61  
 sysop, 3  
 sysop name, 9  
 SYSOPMENU, 57

**T**

TaskView, 7  
 Telebit TrailBlazer, 135  
 TEMP path, 9  
 temporary file path, 86  
 terse log, 6  
 throughput, 124  
 time format, 40  
 TIMEWARN, 57  
 TOOSLOW, 58  
 trace log, 6  
 TTY, 76  
 twit, 3

**U**

UART, 14, 129, 133  
 UDS, 47, 53, 80  
 underscore, 76  
 United Kingdom, 62  
 Unix, 23  
 update attribute, 25  
 upload  
   area, 73  
   log, 58

message, 74  
 protocol, 39  
 ratio, 72  
 temporary path, 86  
 US Robotics, 13, 17  
 user  
   address, 42  
   alias, 43, 44  
   baud, 36, 47, 58  
   birthday, 43  
   city, 43, 83  
   demote, 38  
   download limit, 37  
   expiration, 38, 58, 95  
   hand holding, 44  
   language, 83  
   no password, 46  
   password, 49  
   phone number, 43  
   questionnaire, 45  
   time limit, 36  
 user file, 10  
 user list, 37  
 user-defined section, 47, 53  
 USER.DAT, 83, 109  
 USER.NDX, 83  
 USERNDX.EXE, 83  
 USR\_ADDRESS, 83  
 USR\_ALIAS, 83  
 USR\_ANSWER, 83  
 USR\_BIRTHDAY, 83  
 USR\_CITY, 83  
 USR\_LANG, 83  
 USR\_LST, 83  
 USR\_NAME, 83  
 USR\_PHONE, 83  
 USR\_PWD, 83  
 utilities, 137  
 UUCP, 23, 26, 87  
 UUCPlist, 33

**V**

Valentyn, John, iii  
 verbose log, 6  
 VFOS\_IBM, 129  
 video, 7  
 virus scanning, 36

**W**

Wagner III, Wynn, 96, 121, 126  
 wait, 120  
 watchdog, 9  
 WELCOME, 56, 58  
 words



dirty, 55  
words, dirty, 126  
worthy, 3

## **X**

X00, 129, 133  
XDGONE, 58  
XDWARN, 58  
XFERBAUD, 59  
xmit, 120  
XModem, v, 110  
XON, 17  
XTGONE, 59  
XTWARN, 59

## **Y**

YELL, 59  
YModem, 110

## **Z**

ZIP, 86  
ZIPmail, 19  
ZModem, 82  
zone, 18, 31  
ZOO.EXE, 19  
ZOOmail, 19

